Multiple Context-free Grammars

Course 2: expressive power and parsing

Sylvain Salvati

INRIA Bordeaux Sud-Ouest

ESSLLI 2011

Yesterday

- the relation of MCFG to the formalization of natural language,
- the convergence of many formalisms to the class of MCFL,
- a view of some advantages of algebraicity/context-freeness of grammatical formalisms.

Yesterday

- the relation of MCFG to the formalization of natural language,
- the convergence of many formalisms to the class of MCFL,
- a view of some advantages of algebraicity/context-freeness of grammatical formalisms.

Yesterday

- the relation of MCFG to the formalization of natural language,
- the convergence of many formalisms to the class of MCFL,
- a view of some advantages of algebraicity/context-freeness of grammatical formalisms.

Outline

Definition reminder

A m-MCFG(r) is a 4-tuple (N, T, P, S) such that:

- N is a ranked alphabet of *non-terminals* of max. rank m.
- T is an alphabet of *terminals*
- P is a set of rules of the form:

$$A(s_1, \ldots, s_k) \leftarrow B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, B_n(x_1^n, \ldots, x_{k_n}^n)$$

  where:
  - A is a non-terminal of rank k, B_i is non-terminal of rank k_i, $n \leq r$,
  - the variables x_j^i are pairwise distinct,
  - the strings s_i are in $(T \cup X)^*$ with $X = \bigcup_{i=1}^{n} \bigcup_{j=1}^{k_i} \{x_j^i\}$,
  - each variable x_j^i has at most one occurrence in $s_1 \ldots s_k$
- S is a non-terminal of rank 1, *the starting symbol*.

Definition reminder

A m-MCFG(r) is a 4-tuple (N, T, P, S) such that:

- N is a ranked alphabet of *non-terminals* of max. rank m.
- T is an alphabet of *terminals*
- P is a set of rules of the form:

$$A(s_1, \dots, s_k) \leftarrow B_1(x_1^1, \dots, x_{k_1}^1), \dots, B_n(x_1^n, \dots, x_{k_n}^n)$$

  where:
  - A is a non-terminal of rank k, B_i is non-terminal of rank k_i, $n \leq r$,
  - the variables x_j^i are pairwise distinct,
  - the strings s_i are in $(T \cup X)^*$ with $X = \bigcup_{i=1}^{n} \bigcup_{j=1}^{k_i} \{x_j^i\}$,
  - each variable x_j^i has at most one occurrence in $s_1 \dots s_k$
- S is a non-terminal of rank 1, *the starting symbol*.

Definition reminder

A m-MCFG(r) is a 4-tuple (N, T, P, S) such that:

- N is a ranked alphabet of *non-terminals* of max. rank m.
- T is an alphabet of *terminals*
- P is a set of rules of the form:

$$A(s_1, \ldots, s_k) \leftarrow B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, B_n(x_1^n, \ldots, x_{k_n}^n)$$

  where:
  - A is a non-terminal of rank k, B_i is non-terminal of rank k_i, $n \leq r$,
  - the variables x_j^i are pairwise distinct,
  - the strings s_i are in $(T \cup X)^*$ with $X = \bigcup_{i=1}^{n} \bigcup_{j=1}^{k_i} \{x_j^i\}$,
  - each variable x_j^i has at most one occurrence in $s_1 \ldots s_k$
- S is a non-terminal of rank 1, *the starting symbol*.

Definition reminder

A m-MCFG(r) is a 4-tuple (N, T, P, S) such that:

- N is a ranked alphabet of *non-terminals* of max. rank m.
- T is an alphabet of *terminals*
- P is a set of rules of the form:

$$A(s_1, \ldots, s_k) \leftarrow B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, B_n(x_1^n, \ldots, x_{k_n}^n)$$

  where:
  - A is a non-terminal of rank k, B_i is non-terminal of rank k_i, $n \leq r$,
  - the variables x_j^i are pairwise distinct,
  - the strings s_i are in $(T \cup X)^*$ with $X = \bigcup_{i=1}^{n} \bigcup_{j=1}^{k_i} \{x_j^i\}$,
  - each variable x_j^i has at most one occurrence in $s_1 \ldots s_k$
- S is a non-terminal of rank 1, *the starting symbol*.

The language generated by an MCFG

Given an MCFG $G = (N, T, P, S)$, if the following conditions holds:

- $B_1(s^1_1, \dots, s^1_{k_1}), \dots, B_n(s^n_1, \dots, s^n_{k_n})$ are derivable,
- $A(s_1, \dots, s_k) \leftarrow B_1(x^1_1, \dots, x^1_{k_1}), \dots, B_n(x^n_1, \dots, x^n_{k_n})$ is a rule in P

then $A(t_1, \dots, t_k)$ with $t_i = s_i[x^i_j \leftarrow s^i_j]_{i \in [1;n], j \in [1;k_i]}$ is derivable.

The language define by G, $L(G)$ is:

$$\{w \mid S(w) \text{ is derivable}\}$$

The language generated by an MCFG

Given an MCFG $G = (N, T, P, S)$, if the following conditions holds:

- $B_1(s^1_1, \ldots, s^1_{k_1}), \ldots, B_n(s^n_1, \ldots, s^n_{k_n})$ are derivable,
- $A(s_1, \ldots, s_k) \leftarrow B_1(x^1_1, \ldots, x^1_{k_1}), \ldots, B_n(x^n_1, \ldots, x^n_{k_n})$ is a rule in P

then $A(t_1, \ldots, t_k)$ with $t_i = s_i[x^i_j \leftarrow s^i_j]_{i \in [1;n], j \in [1;k_i]}$ is derivable.

The language define by G, $L(G)$ is:

$$\{w \mid S(w) \text{ is derivable}\}$$

Outline

The two parameters

In m-MCFL(r):

- we call m the *dimension*
- we call r the *rank*

We can trade rank for dimension:

$$
\begin{array}{rcl}
A(s_1,\ldots,s_m) & \leftarrow & B_1(x_1^1,\ldots,x_{k_1}^1),\ \ldots,\\
 & & B_{n+1}(x_1^{n+1},\ldots,x_{k_{n+1}}^{n+1})\\
 & \Downarrow & \\
A(s_1,\ldots,s_m) & \leftarrow & B_1(x_1^1,\ldots,x_{k_1}^1),\ \ldots,\\
 & & C(x_1^n,\ldots,x_{k_n}^n,x_1^{n+1},\ldots,x_{k_{n+1}}^{n+1})\\
C(x_1^n,\ldots,x_{k_n}^n,x_1^{n+1},\ldots,x_{k_{n+1}}^{n+1}) & \leftarrow & B_n(x_1^n,\ldots,x_{k_n}^n),\\
 & & B_{n+1}(x_1^{n+1},\ldots,x_{k_{n+1}}^{n+1})
\end{array}
$$

m-MCFL(r) $\supseteq$ ($m \times (k+1)$)-MCFL($r-k$) when $1 \leq k \leq r-2$

The two parameters

In m-MCFL(r):

- we call m the *dimension*
- we call r the *rank*

We can trade rank for dimension:

$$
\begin{array}{rcl}
A(s_1, \ldots, s_m) & \leftarrow & B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, \\
& & B_{n+1}(x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1}) \\
& \Downarrow & \\
A(s_1, \ldots, s_m) & \leftarrow & B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, \\
& & C(x_1^n, \ldots, x_{k_n}^n, x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1}) \\
C(x_1^n, \ldots, x_{k_n}^n, x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1}) & \leftarrow & B_n(x_1^n, \ldots, x_{k_n}^n), \\
& & B_{n+1}(x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1})
\end{array}
$$

$$m\text{-MCFL}(r) \supseteq (m \times (k+1))\text{-MCFL}(r-k) \text{ when } 1 \leq k \leq r-2$$

The two parameters

In m-MCFL(r):

- we call m the *dimension*
- we call r the *rank*

We can trade rank for dimension:

$$\begin{array}{rcl}
A(s_1, \ldots, s_m) & \leftarrow & B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, \\
 & & B_{n+1}(x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1}) \\
 & \Downarrow & \\
A(s_1, \ldots, s_m) & \leftarrow & B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, \\
 & & C(x_1^n, \ldots, x_{k_n}^n, x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1}) \\
C(x_1^n, \ldots, x_{k_n}^n, x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1}) & \leftarrow & B_n(x_1^n, \ldots, x_{k_n}^n), \\
 & & B_{n+1}(x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1})
\end{array}$$

m-MCFL(r) $\supseteq$ $(m \times (k+1))$-MCFL($r-k$) when $1 \leq k \leq r-2$

The two parameters

In m-MCFL(r):

- we call m the *dimension*
- we call r the *rank*

We can trade rank for dimension:

$$\begin{array}{rcl}
A(s_1, \ldots, s_m) & \leftarrow & B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, \\
 & & B_{n+1}(x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1}) \\
 & \Downarrow & \\
A(s_1, \ldots, s_m) & \leftarrow & B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, \\
 & & C(x_1^n, \ldots, x_{k_n}^n, x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1}) \\
C(x_1^n, \ldots, x_{k_n}^n, x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1}) & \leftarrow & B_n(x_1^n, \ldots, x_{k_n}^n), \\
 & & B_{n+1}(x_1^{n+1}, \ldots, x_{k_{n+1}}^{n+1})
\end{array}$$

$$m\text{-MCFL}(r) \supseteq (m \times (k+1))\text{-MCFL}(r-k) \text{ when } 1 \leq k \leq r-2$$

The two parameters

We cannot trade dimension for rank.

Theorem (Seki *et al.*)

The language $L_{m+1} = \{a_1^n b_1^n \dots a_{m+1}^n b_{m+1}^n \mid n \in \mathbb{N}\}$ *is a* $(m+1)$*-MCFL*(1) *but is not a* m*-MCFL*(r) *for any* r

Proof.

The two parameters

We cannot trade dimension for rank.

Theorem (Seki *et al.*)

The language $L_{m+1} = \{a_1^n b_1^n \ldots a_{m+1}^n b_{m+1}^n \mid n \in \mathbb{N}\}$ *is a* $(m+1)$*-MCFL*(1) *but is not a* m*-MCFL*(r) *for any* r

Proof.

L_{m+1} is a $(m+1)$-MCFL(1):

$$\begin{aligned} S(x_1 \ldots x_{m+1}) &\leftarrow A(x_1, \ldots, x_{m+1}) \\ A(a_1 x_1 b_1, \ldots, a_m x_m b_m) &\leftarrow A(x_1, \ldots, x_{m+1}) \\ A(\epsilon, \ldots, \epsilon) &\leftarrow \end{aligned}$$

□

The two parameters

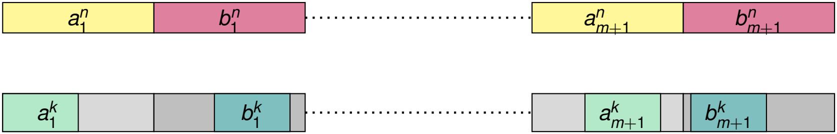
We cannot trade dimension for rank.

Theorem (Seki *et al.*)

The language $L_{m+1} = \{a_1^n b_1^n \dots a_{m+1}^n b_{m+1}^n \mid n \in \mathbb{N}\}$ *is a* $(m+1)$*-MCFL*(1) *but is not a* m*-MCFL*(r) *for any* r

Proof.

$L_{m+1} \notin m$-MCFL(r) based on a weak form of pumping lemma: there is a string that is $2m$-pumpable/iterable.



□

Outline

Two hierarchies

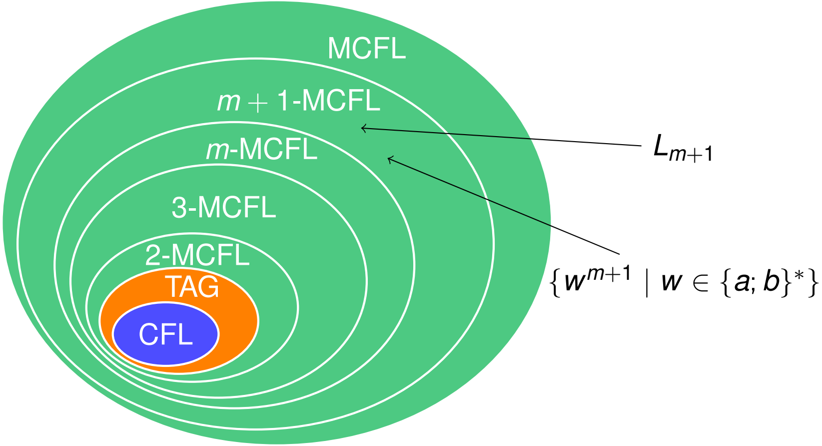
Since:

$$\mathrm{MCFL}(r+1) = \mathrm{MCFL}(r) \text{ when } r \geq 2$$

We are interested in two hierarchies:

- the hierarchy indexed by m of m-MCFL = $\bigcup_{r\in\mathbb{N}} m$-MCFL$(r)$
- the hierarchy indexed by r of m-MCFL(r)

An infinite hierarchy with respect to the dimension



The hierarchy of 1-MCFL(r)

Theorem

$$1\text{-}MCFL = CFL$$

Proof.

$$\begin{array}{ccc} A(s) & \leftarrow & B_1(x_1), \ldots, B_n(x_n) \\ & \Downarrow & \\ A & \rightarrow & s[x_i := B_i]_{i=1}^{n} \end{array}$$

□

- 1-MCFL(1) $\subsetneq$ 1-MCFL(2)
- 1-MCFL(r) $=$ 1-MCFL($r+1$) when $r \geq 2$

The hierarchy of 1-MCFL(r)

Theorem

$$1\text{-}MCFL = CFL$$

Proof.

$$\begin{array}{rcl} A(s) & \leftarrow & B_1(x_1), \ldots, B_n(x_n) \\ & \Downarrow & \\ A & \rightarrow & s[x_i := B_i]_{i=1}^{n} \end{array}$$

$\square$

- 1-MCFL(1) $\subsetneq$ 1-MCFL(2)
- 1-MCFL(r) $=$ 1-MCFL($r+1$) when $r \geq 2$

2-MCFL(2) and 2-MCFL(3)

Theorem (Rambow,Satta 1994)

$2\text{-}MCFL(2) = 2\text{-}MCFL(3)$

Proof.

$$
\begin{array}{rcl}
A(u_1 x_1 u_2 y_2 u_3 z_1 u_4, v_1 y_1 v_2 x_2 v_3 z_2 v_4) & \leftarrow & B_1(x_1, x_2),\ B_2(y_1, y_2), \\
 & & B_3(z_1, z_2) \\
 & \Downarrow & \\
A(x_1 z_1 u_4, x_2 z_2 v_4) & \leftarrow & B_{1,2}(x_1, x_2),\ B_3(z_1, z_2) \\
B_{1,2}(u_1 x_1 u_2 y_2 u_3, v_1 y_1 v_2 x_2 v_3) & \leftarrow & B_1(x_1, x_2),\ B_2(y_1, y_2)
\end{array}
$$

$\square$

2-MCFL(2) and 2-MCFL(3)

Theorem (Rambow,Satta 1994)

$2\text{-}MCFL(2) = 2\text{-}MCFL(3)$

Proof.

$$A(u_1 x_1 u_2 y_2 u_3 z_1 u_4, v_1 y_1 v_2 x_2 v_3 z_2 v_4) \leftarrow B_1(x_1, x_2), B_2(y_1, y_2), B_3(z_1, z_2)$$

$$\Downarrow$$

$$A(x_1 z_1 u_4, x_2 z_2 v_4) \leftarrow B_{1,2}(x_1, x_2), B_3(z_1, z_2)$$

$$B_{1,2}(u_1 x_1 u_2 y_2 u_3, v_1 y_1 v_2 x_2 v_3) \leftarrow B_1(x_1, x_2), B_2(y_1, y_2)$$

$\square$

2-MCFL(2) and 2-MCFL(3)

Theorem (Rambow,Satta 1994)

$2\text{-}MCFL(2) = 2\text{-}MCFL(3)$

Proof.

$$A(u_1 x_1 u_2 y_2 u_3 z_1 u_4, v_1 y_1 v_2 x_2 v_3 z_2 v_4) \leftarrow B_1(x_1, x_2),\ B_2(y_1, y_2),\ B_3(z_1, z_2)$$

$$\Downarrow$$

$$A(x_1 z_1 u_4, x_2 z_2 v_4) \leftarrow B_{1,2}(x_1, x_2),\ B_3(z_1, z_2)$$

$$B_{1,2}(u_1 x_1 u_2 y_2 u_3, v_1 y_1 v_2 x_2 v_3) \leftarrow B_1(x_1, x_2),\ B_2(y_1, y_2)$$

□

The language $L_{r,m,\sigma}$

Let $\Sigma_{i,r} = \{a_{i,1}, \ldots, a_{i,r}\}$, $w_{\mathbf{v},i,\sigma} = a_{i,\sigma(1)}^{n_{\sigma(1)}} \ldots a_{i,\sigma(r)}^{n_{\sigma(r)}}$ with $\mathbf{v} = (n_1, \ldots, n_r) \in \mathbb{N}^r$ and σ permutation of $[1; r]$.

Theorem (Rambow,Satta 1994)

If $\sigma(1) = 1$, $\sigma(r) = r$, $\sigma(i+1) \neq \sigma(i) + 1$ and $r \geq 3$, the language $L_{r+3,m,\sigma} = \{w_{\mathbf{v},1,Id} w_{\mathbf{v},2,\sigma} \ldots w_{\mathbf{v},m,\sigma} \mid \mathbf{v} \in \mathbb{N}^r\}$ is an m-MCFL$(r+1)$ but not an m-MCFL(r).

Theorem

If $m > 2$ or $r > 2$ then

$$m\text{-}MCFL(r) \subsetneq m\text{-}MCFL(r+1)$$

The language $L_{r,m,\sigma}$

Let $\Sigma_{i,r} = \{a_{i,1}, \ldots, a_{i,r}\}$, $w_{\mathbf{v},i,\sigma} = a_{i,\sigma(1)}^{n_{\sigma(1)}} \ldots a_{i,\sigma(r)}^{n_{\sigma(r)}}$ with $\mathbf{v} = (n_1, \ldots, n_r) \in \mathbb{N}^r$ and σ permutation of $[1; r]$.

Theorem (Rambow,Satta 1994)

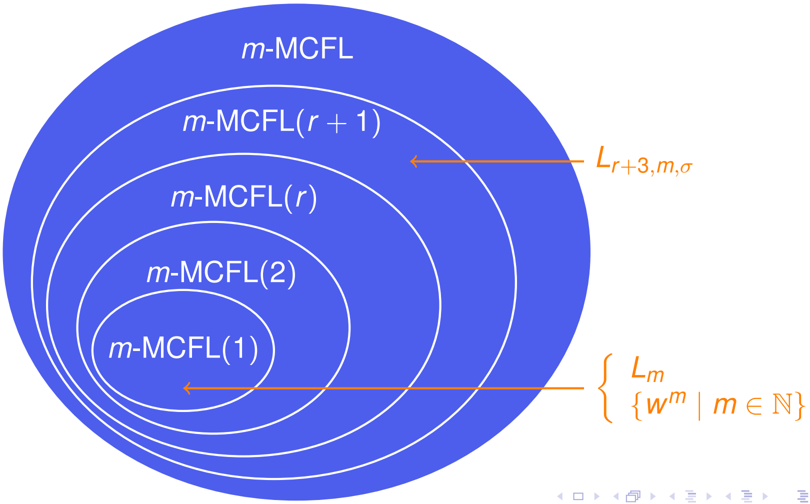
If $\sigma(1) = 1$, $\sigma(r) = r$, $\sigma(i+1) \neq \sigma(i) + 1$ and $r \geq 3$, the language $L_{r+3,m,\sigma} = \{w_{\mathbf{v},1,Id} w_{\mathbf{v},2,\sigma} \ldots w_{\mathbf{v},m,\sigma} \mid \mathbf{v} \in \mathbb{N}^r\}$ is an m-MCFL$(r+1)$ but not an m-MCFL(r).

Theorem

If $m > 2$ or $r > 2$ then

$$m\text{-}MCFL(r) \subsetneq m\text{-}MCFL(r+1)$$

An infinite hierarchy with respect to the rank

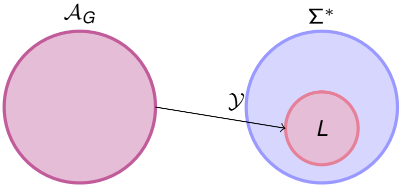


Outline

Outline

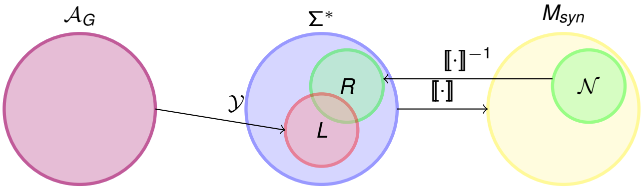
m-MCFL(r) are closed under intersection with regular sets

Given an m-MCFL(r) G and a regular set R



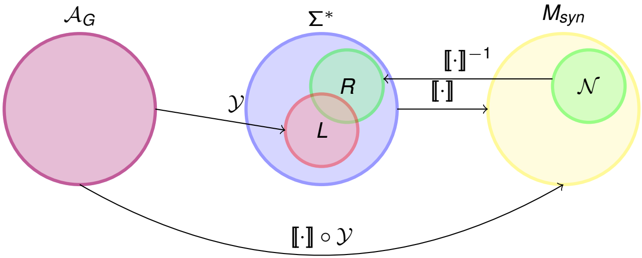
m-MCFL(r) are closed under intersection with regular sets

Given an m-MCFL(r) G and a regular set R



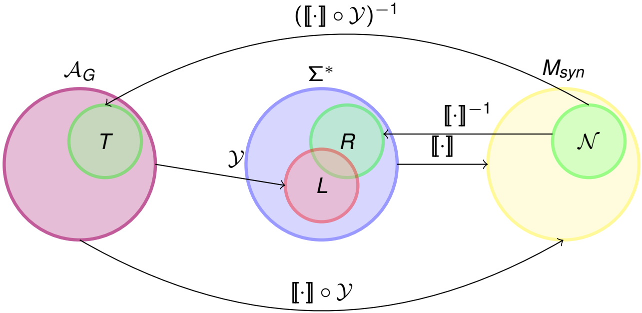
m-MCFL(r) are closed under intersection with regular sets

Given an m-MCFL(r) G and a regular set R



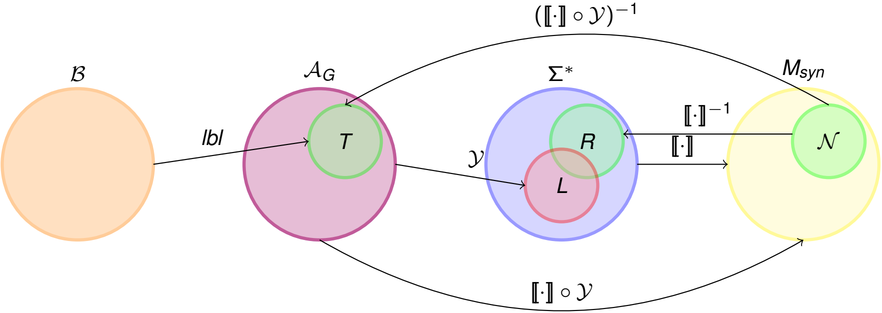
m-MCFL(r) are closed under intersection with regular sets

Given an m-MCFL(r) G and a regular set R



m-MCFL(r) are closed under intersection with regular sets

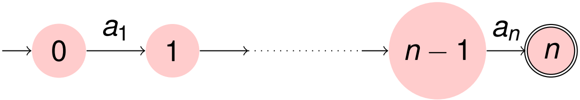
Given an m-MCFL(r) G and a regular set R



The grammar we obtain uses rules that are similar to those of G.

Singleton sets are regular

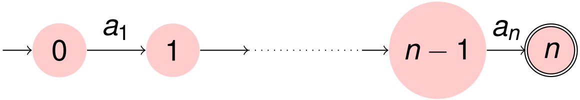
Given a word $a_1 \dots a_n$ it is recognized by the following automaton:



- We get the decidability of the membership problem for MCFGs.
- NB: a similar technique yields decidability of the generation problem.

Singleton sets are regular

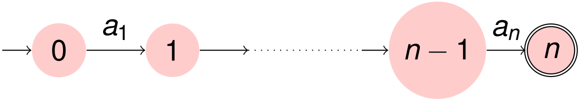
Given a word $a_1 \ldots a_n$ it is recognized by the following automaton:



- We get the decidability of the membership problem for MCFGs.
- NB: a similar technique yields decidability of the generation problem.

Singleton sets are regular

Given a word $a_1 \ldots a_n$ it is recognized by the following automaton:



- We get the decidability of the membership problem for MCFGs.
- NB: a similar technique yields decidability of the generation problem.

Outline

A direct construction of the tree automaton

Suppose that R is given by the automaton (Q, δ, q_0, Q_f), then the states of the tree automaton are the tuples

$$(q_1, q'_1, \ldots, q_k, q'_k)$$

such that:

- q_i and q'_i are in Q
- k is the rank of a predicate of the MCFG

A direct construction of the tree automaton

Suppose that R is given by the automaton (Q, δ, q_0, Q_f), then the states of the tree automaton are the tuples

$$(q_1, q_1', \dots, q_k, q_k')$$

such that:

- q_i and q_i' are in Q
- k is the rank of a predicate of the MCFG

Given a rule

$$\pi = A(s_1, \dots, s_n) \leftarrow B_1(x_1^1, \dots x_{k_1}^1), \dots, B_m(x_1^m, \dots, x_{k_m}^m)$$

We built tree automaton rules as

$$\pi(q_1^1, p_1^1, \dots q_{k_1}^1, p_{k_1}^1) \dots (q_1^m, p_1^m, \dots q_{k_m}^m, p_{k_m}^m) \rightarrow (q_1, p_1, \dots, q_n, p_n)$$

if $\delta'(s_i, q_i) = p_i$ with $\delta'(a, q) = \delta(a, q)$ and $\delta'(x_i^j, q_i^j) = p_i^j$.

A direct construction of the tree automaton

Suppose that R is given by the automaton (Q, δ, q_0, Q_f), then the states of the tree automaton are the tuples

$$(q_1, q'_1, \ldots, q_k, q'_k)$$

such that:

- q_i and q'_i are in Q
- k is the rank of a predicate of the MCFG

Therefore the size of the tree automaton is a $\mathcal{O}(|Q|^{2m(r+1)})$ and universal membership of m-MCFG(r) can be performed in polynomial time.

Complexity when on m or r is fixed

- Satta (1992)

Theorem

- *the membership problem for MFCG(1) is NP-complete,*
- *the membership problem for 2-MCFG is NP-hard,*
- *the membership problem for non-erasing m-MCFG is in NP,*
- *Kaji, Nakanishi, Seki and Kasami (1994) proved that the membership problem for non-erasing m-MCFG is in NP.*

The X3C problem

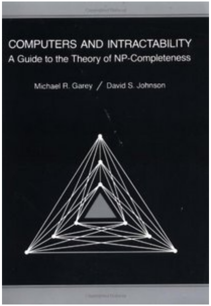
INPUT $X = \{a_1; \ldots a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1; m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

The X3C problem

INPUT $X = \{a_1; \ldots a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$
OUTPUT YES when there is $P \subseteq [1;m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

X3C is NP-complete.



The X3C problem

INPUT $X = \{a_1; \dots a_{3n}\}$ $(B_k)_{i\in[1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1;m]$ such that $(B_k)_{k\in P}$ is a partition of X; NO otherwise

$X = \{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$

$B_1 = \{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\}$ $B_2 = \{\boxed{a_3}; \boxed{a_6}; \boxed{a_9}\}$

$B_3 = \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\}$ $B_4 = \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$

A solution is $\{B_1; B_3; B_4\}$ since

$$\{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\} \cup \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\} \cup \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$$

$$=$$

$$\{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$$

The X3C problem

INPUT $X = \{a_1; \ldots a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1;m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

$X = \{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$

$B_1 = \{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\}$ $B_2 = \{\boxed{a_3}; \boxed{a_6}; \boxed{a_9}\}$

$B_3 = \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\}$ $B_4 = \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$

A solution is $\{B_1; B_3; B_4\}$ since

$$\{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\} \cup \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\} \cup \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$$

$$=$$

$$\{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$$

The X3C problem

INPUT $X = \{a_1; \dots a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1; m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

$X = \{a_1; a_2; a_3; a_4; a_5; a_6; a_7; a_8; a_9\}$

$B_1 = \{a_4; a_7; a_9\}$ $B_2 = \{a_3; a_6; a_9\}$

$B_3 = \{a_1; a_3; a_8\}$ $B_4 = \{a_2; a_5; a_6\}$

A solution is $\{B_1; B_3; B_4\}$ since

$$\{a_4; a_7; a_9\} \cup \{a_1; a_3; a_8\} \cup \{a_2; a_5; a_6\}$$

$$=$$

$$\{a_1; a_2; a_3; a_4; a_5; a_6; a_7; a_8; a_9\}$$

NP-hardness of the membership problem for MCFG(1)

The X3C problem:

INPUT $X = \{a_1; \ldots a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1;m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

An $MCFG(1)$ for an instance of X3C:
The word is $a_1 \ldots a_{3n}$

$$
\begin{array}{rl}
S(x_1 \ldots x_{3n}) \leftarrow & A_0(x_1, \ldots, x_{3n}) \\
A_l(s_1, \ldots, s_{3n}) \leftarrow & A_{l+1}(x_1, \ldots, x_{3n}) \text{ for } k \in [1;m],\ B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\} \\
 & s_{i_j} = \begin{cases} s_i = x_i a_{k_j} \text{ if } i = k_j \\ s_i = x_i \text{ otherwise} \end{cases} \\
A_m(\epsilon, \ldots, \epsilon) \leftarrow &
\end{array}
$$

Note that the language is finite.

NP-hardness of the membership problem for MCFG(1)

The X3C problem:

INPUT $X = \{a_1; \dots a_{3n}\}$ $(B_k)_{i\in[1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1; m]$ such that $(B_k)_{k\in P}$ is a partition of X; NO otherwise

$X = \{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$

$B_1 = \{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\}$ $B_2 = \{\boxed{a_3}; \boxed{a_6}; \boxed{a_9}\}$

$B_3 = \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\}$ $B_4 = \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$

$$A_3(\epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon)$$

NP-hardness of the membership problem for MCFG(1)

The X3C problem:

INPUT $X = \{a_1; \ldots a_{3n}\}$ $(B_k)_{i\in[1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1; m]$ such that $(B_k)_{k\in P}$ is a partition of X; NO otherwise

$X = \{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$

$B_1 = \{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\}$ $B_2 = \{\boxed{a_3}; \boxed{a_6}; \boxed{a_9}\}$

$B_3 = \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\}$ $B_4 = \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$

$$\frac{A_3(\epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon)}{A_2(\epsilon, \epsilon, \epsilon, \boxed{a_4}, \epsilon, \epsilon, \boxed{a_7}, \epsilon, \boxed{a_9})}$$

NP-hardness of the membership problem for MCFG(1)

The X3C problem:

INPUT $X = \{a_1; \dots a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1;m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

$X = \{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$

$B_1 = \{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\}$ $B_2 = \{\boxed{a_3}; \boxed{a_6}; \boxed{a_9}\}$

$B_3 = \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\}$ $B_4 = \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$

$$\dfrac{\dfrac{A_3(\epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon)}{A_2(\epsilon, \epsilon, \epsilon, \boxed{a_4}, \epsilon, \epsilon, \boxed{a_7}, \epsilon, \boxed{a_9})}}{A_1(\boxed{a_1}, \epsilon, \boxed{a_3}, \boxed{a_4}, \epsilon, \epsilon, \boxed{a_7}, \boxed{a_8}, \boxed{a_9})}$$

NP-hardness of the membership problem for MCFG(1)

The X3C problem:

INPUT $X = \{a_1; \dots a_{3n}\}$ $(B_k)_{i\in[1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1; m]$ such that $(B_k)_{k\in P}$ is a partition of X; NO otherwise

$X = \{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$

$B_1 = \{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\}$ $B_2 = \{\boxed{a_3}; \boxed{a_6}; \boxed{a_9}\}$

$B_3 = \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\}$ $B_4 = \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$

$$\dfrac{\dfrac{\dfrac{A_3(\epsilon,\epsilon,\epsilon,\epsilon,\epsilon,\epsilon,\epsilon,\epsilon,\epsilon)}{A_2(\epsilon,\epsilon,\epsilon,\boxed{a_4},\epsilon,\epsilon,\boxed{a_7},\epsilon,\boxed{a_9})}}{A_1(\boxed{a_1},\epsilon,\boxed{a_3},\boxed{a_4},\epsilon,\epsilon,\boxed{a_7},\boxed{a_8},\boxed{a_9})}}{A_0(\boxed{a_1},\boxed{a_2},\boxed{a_3},\boxed{a_4},\boxed{a_5},\boxed{a_6},\boxed{a_7},\boxed{a_8},\boxed{a_9})}$$

NP-hardness of the membership problem for MCFG(1)

The X3C problem:

INPUT $X = \{a_1; \dots a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1; m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

$X = \{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$

$B_1 = \{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\}$ $B_2 = \{\boxed{a_3}; \boxed{a_6}; \boxed{a_9}\}$

$B_3 = \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\}$ $B_4 = \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$

$$\begin{array}{c}
A_3(\epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon) \\ \hline
A_2(\epsilon, \epsilon, \epsilon, \boxed{a_4}, \epsilon, \epsilon, \boxed{a_7}, \epsilon, \boxed{a_9}) \\ \hline
A_1(\boxed{a_1}, \epsilon, \boxed{a_3}, \boxed{a_4}, \epsilon, \epsilon, \boxed{a_7}, \boxed{a_8}, \boxed{a_9}) \\ \hline
A_0(\boxed{a_1}, \boxed{a_2}, \boxed{a_3}, \boxed{a_4}, \boxed{a_5}, \boxed{a_6}, \boxed{a_7}, \boxed{a_8}, \boxed{a_9}) \\ \hline
S(\boxed{a_1}\boxed{a_2}\boxed{a_3}\boxed{a_4}\boxed{a_5}\boxed{a_6}\boxed{a_7}\boxed{a_8}\boxed{a_9})
\end{array}$$

NP-hardness of the membership problem for 3-MCFG

The X3C problem:

INPUT $X = \{a_1; \ldots; a_{3n}\}$ $(B_k)_{i\in[1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1; m]$ such that $(B_k)_{k\in P}$ is a partition of X; NO otherwise

A 3-MCFG for an instance of X3C:
The word is $a_1 \ldots a_{3n}$

$$\begin{array}{rcl}
B_k(a_{k_1}, a_{k_2}, a_{k_3}) & \leftarrow & \\
B_k(\epsilon, \epsilon, \epsilon) & \leftarrow & \\
S(s_1 \ldots s_{3n}) & \leftarrow & B_1(y_{1,1}, y_{1,2}, y_{1,3}), \ \ldots, \ B_m(y_{m,1}, y_{m,2}, y_{m,3})
\end{array}$$

With $s_i = y_{p_1,l_1} \ldots y_{p_r,l_r}$ so that $B_{p_j} = \{a_{p_j,1}; a_{p_j,2}; a_{p_j,3}\}$ and $a_{p_j,l_j} = a_i$.
NB: again the language is finite.

NP-hardness of the membership problem for 3-MCFG

The X3C problem:

INPUT $X = \{a_1; \ldots; a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1; m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

$X = \{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$

$B_1 = \{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\}$ $B_2 = \{\boxed{a_3}; \boxed{a_6}; \boxed{a_9}\}$

$B_3 = \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\}$ $B_4 = \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$

$$\frac{B_1(\boxed{a_4}, \boxed{a_7}, \boxed{a_9})\ B_2(\epsilon, \epsilon, \epsilon)\ B_3(\boxed{a_1}, \boxed{a_3}, \boxed{a_8})\ B_4(\boxed{a_2}, \boxed{a_5}, \boxed{a_6})}{S(\boxed{a_1}\boxed{a_2}\boxed{a_3}\boxed{a_4}\boxed{a_5}\boxed{a_6}\boxed{a_7}\boxed{a_8}\boxed{a_9})}$$

NP-hardness of the membership problem for 2-MCFG

The X3C problem:

INPUT $X = \{a_1; \ldots; a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1;m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

A 2-MCFG for an instance of X3C:
The word is $(\alpha\beta)^m a_1 \ldots a_{3n}$

$$
\begin{array}{r||r||c}
B_{k,1}(\alpha, a_{k_1}) \leftarrow & B_{k,1}(\epsilon, \epsilon) \leftarrow & S(x_{1,1}x_{1,2}x_{1,3}\ldots x_{m,1}x_{m,2}x_{m,3}s_1\ldots s_{3n}) \\
B_{k,2}(\epsilon\, a_{k_2}) \leftarrow & B_{k,2}(\alpha\beta, \epsilon) \leftarrow & \leftarrow \\
B_{k,3}(\beta, a_{k_3}) \leftarrow & B_{k,3}(\epsilon, \epsilon) \leftarrow & B_{1,1}(x_{1,1}, y_{1,1}),\ B_{1,2}(x_{1,2}, y_{1,2}),\ B_{1,3}(x_{1,3}, y_{1,3}) \\
 & & \vdots \\
 & & B_{k,1}(x_{1,1}, y_{k,1}),\ B_{k,2}(x_{1,2}, y_{k,2}),\ B_{k,3}(x_{1,3}, y_{k,3})
\end{array}
$$

With $s_i = y_{p_1,l_1}\ldots y_{p_r,l_r}$ so that $B_{p_j} = \{a_{p_j,1}; a_{p_j,2}; a_{p_j,3}\}$ and $a_{p_j,l_j} = a_i$.
NB: again the language is finite.

NP-hardness of the membership problem for 2-MCFG

The X3C problem:

INPUT $X = \{a_1; \ldots; a_{3n}\}$ $(B_k)_{i\in[1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1;m]$ such that $(B_k)_{k\in P}$ is a partition of X; NO otherwise

A 2-MCFG for an instance of X3C:
The word is $(\alpha\beta)^m a_1 \ldots a_{3n}$

$$\begin{array}{r||r} B_{k,1}(\alpha, a_{k_1}) \leftarrow & B_{k,1}(\epsilon,\epsilon) \leftarrow \\ B_{k,2}(\epsilon\, a_{k_2}) \leftarrow & B_{k,2}(\alpha\beta,\epsilon) \leftarrow \\ B_{k,3}(\beta, a_{k_3}) \leftarrow & B_{k,3}(\epsilon,\epsilon) \leftarrow \end{array}$$

		$x_{k,1}x_{k,2}x_{k,3}$			$x_{k,1}x_{k,2}x_{k,3}$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon,\epsilon) \leftarrow$ $B_{k,2}(\alpha\beta,\epsilon) \leftarrow$ $B_{k,3}(\epsilon,\epsilon) \leftarrow$	$\alpha\beta$	$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon,\epsilon) \leftarrow$ $B_{k,2}(\alpha\beta,\epsilon) \leftarrow$ $B_{k,3}(\epsilon,\epsilon) \leftarrow$	$\alpha\beta$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon,\epsilon) \leftarrow$ $B_{k,2}(\alpha\beta,\epsilon) \leftarrow$ $B_{k,3}(\epsilon,\epsilon) \leftarrow$	$\alpha\alpha\beta\beta$	$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon,\epsilon) \leftarrow$ $B_{k,2}(\alpha\beta,\epsilon) \leftarrow$ $B_{k,3}(\epsilon,\epsilon) \leftarrow$	ϵ
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon,\epsilon) \leftarrow$ $B_{k,2}(\alpha\beta,\epsilon) \leftarrow$ $B_{k,3}(\epsilon,\epsilon) \leftarrow$	$\alpha\alpha$	$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon,\epsilon) \leftarrow$ $B_{k,2}(\alpha\beta,\epsilon) \leftarrow$ $B_{k,3}(\epsilon,\epsilon) \leftarrow$	β
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon,\epsilon) \leftarrow$ $B_{k,2}(\alpha\beta,\epsilon) \leftarrow$ $B_{k,3}(\epsilon,\epsilon) \leftarrow$	α	$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon,\epsilon) \leftarrow$ $B_{k,2}(\alpha\beta,\epsilon) \leftarrow$ $B_{k,3}(\epsilon,\epsilon) \leftarrow$	$\alpha\beta\beta$

NP-hardness of the membership problem for 2-MCFG

The X3C problem:

INPUT $X = \{a_1; \ldots; a_{3n}\}$ $(B_k)_{i \in [1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1;m]$ such that $(B_k)_{k \in P}$ is a partition of X; NO otherwise

A 2-MCFG for an instance of X3C:
The word is $(\alpha\beta)^m a_1 \ldots a_{3n}$

$$\begin{array}{r||r} B_{k,1}(\alpha, a_{k_1}) \leftarrow & B_{k,1}(\epsilon, \epsilon) \leftarrow \\ B_{k,2}(\epsilon\, a_{k_2}) \leftarrow & B_{k,2}(\alpha\beta, \epsilon) \leftarrow \\ B_{k,3}(\beta, a_{k_3}) \leftarrow & B_{k,3}(\epsilon, \epsilon) \leftarrow \end{array}$$

		$x_{k,1}x_{k,2}x_{k,3}$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\beta$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\alpha\beta\beta$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\alpha$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	α

		$x_{k,1}x_{k,2}x_{k,3}$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\beta$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	ϵ
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	β
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\beta\beta$

NP-hardness of the membership problem for 2-MCFG

The X3C problem:

INPUT $X = \{a_1; \ldots; a_{3n}\}$ $(B_k)_{i\in[1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1;m]$ such that $(B_k)_{k\in P}$ is a partition of X; NO otherwise

A 2-MCFG for an instance of X3C:
The word is $(\alpha\beta)^m a_1 \ldots a_{3n}$

$$\begin{array}{r||r} B_{k,1}(\alpha, a_{k_1}) \leftarrow & B_{k,1}(\epsilon, \epsilon) \leftarrow \\ B_{k,2}(\epsilon\, a_{k_2}) \leftarrow & B_{k,2}(\alpha\beta, \epsilon) \leftarrow \\ B_{k,3}(\beta, a_{k_3}) \leftarrow & B_{k,3}(\epsilon, \epsilon) \leftarrow \end{array}$$

		$x_{k,1}x_{k,2}x_{k,3}$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\beta$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\alpha\beta\beta$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\alpha$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	α

		$x_{k,1}x_{k,2}x_{k,3}$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\beta$
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	ϵ
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	β
$B_{k,1}(\alpha, a_{k_1}) \leftarrow$ $B_{k,2}(\epsilon\, a_{k_2}) \leftarrow$ $B_{k,3}(\beta, a_{k_3}) \leftarrow$	$B_{k,1}(\epsilon, \epsilon) \leftarrow$ $B_{k,2}(\alpha\beta, \epsilon) \leftarrow$ $B_{k,3}(\epsilon, \epsilon) \leftarrow$	$\alpha\beta\beta$

NP-hardness of the membership problem for 2-MCFG

The X3C problem:

INPUT $X = \{a_1; \ldots; a_{3n}\}$ $(B_k)_{i\in[1;m]}$ with $B_k = \{a_{k_1}; a_{k_2}; a_{k_3}\}$

OUTPUT YES when there is $P \subseteq [1; m]$ such that $(B_k)_{k\in P}$ is a partition of X; NO otherwise

A 2-MCFG for an instance of X3C:

The word is $(\alpha\beta)^m a_1 \ldots a_{3n}$

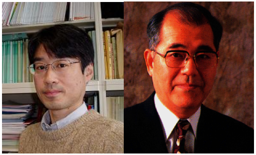
$X = \{\boxed{a_1}; \boxed{a_2}; \boxed{a_3}; \boxed{a_4}; \boxed{a_5}; \boxed{a_6}; \boxed{a_7}; \boxed{a_8}; \boxed{a_9}\}$

$B_1 = \{\boxed{a_4}; \boxed{a_7}; \boxed{a_9}\}$ $B_2 = \{\boxed{a_3}; \boxed{a_6}; \boxed{a_9}\}$

$B_3 = \{\boxed{a_1}; \boxed{a_3}; \boxed{a_8}\}$ $B_4 = \{\boxed{a_2}; \boxed{a_5}; \boxed{a_6}\}$

$$\frac{\begin{array}{l} B_{1,1}(\alpha, \boxed{a_4})\ B_{1,2}(\epsilon, \boxed{a_7})\ B_{1,3}(\beta, \boxed{a_9}) \\ B_{2,1}(\epsilon, \epsilon)\ B_{2,2}(\alpha\beta, \epsilon)\ B_{2,3}(\epsilon, \epsilon) \\ B_{3,1}(\alpha, \boxed{a_1})\ B_{3,2}(\epsilon, \boxed{a_3})\ B_{3,3}(\beta, \boxed{a_8}) \\ B_{4,1}(\alpha, \boxed{a_2})\ B_{4,2}(\epsilon, \boxed{a_5})\ B_{4,3}(\beta, \boxed{a_6}) \end{array}}{S((\alpha\beta)^3 \boxed{a_1}\boxed{a_2}\boxed{a_3}\boxed{a_4}\boxed{a_5}\boxed{a_6}\boxed{a_7}\boxed{a_8}\boxed{a_9})}$$

The membership problem for MCFG



- Kaji, Nakanishi, Seki, Kasami

Theorem

- *the membership problem for non-erasing MCFG is PSPACE-complete,*
- *the membership problem for MCFG is EXPTIME-complete.*

In a nutshell

In general we cannot hope to perform much better than

$$\mathcal{O}(|Q|^{2m(r+1)})$$

Outline

- Expressiveness
  - Two parameters
  - Two hierarchies
- Membership complexity
  - Recognizability and decidability of membership
  - Complexity of the membership problem
- Datalog parsing
  - Earley parsing and datalog
  - Earley parsing for MCFG

Parsing and logic

Claim:

Parsing $\rightarrow$ exploit algebraic structure and its connection to logic.

Outline

Parsing CFG Items

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \ldots a_n$, a rule $A \to \alpha\beta$, an item is:

$$[A \to \alpha \bullet \beta](i, j) \text{ with } i, j \in [1; n]$$

it means that:

- $\alpha \xrightarrow{*} a_{i+1} \ldots a_j$
- $S \xrightarrow{*} a_1 \ldots a_i A\gamma$

Parsing CFG Items

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \dots a_n$, a rule $A \to \alpha\beta$, an item is:

$$[A \to \alpha \bullet \beta](i, j) \text{ with } i, j \in [1; n]$$

it means that:

- $\alpha \xrightarrow{*} a_{i+1} \dots a_j$
- $S \xrightarrow{*} a_1 \dots a_i A \gamma$

Parsing CFG Items

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \dots a_n$, a rule $A \to \alpha\beta$, an item is:

$$[A \to \alpha \bullet \beta](i, j) \text{ with } i, j \in [1; n]$$

it means that:

- $\alpha \xrightarrow{*} a_{i+1} \dots a_j$
- $S \xrightarrow{*} a_1 \dots a_i A\gamma$

An inference system on items

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \dots a_n$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \textit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \textit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ \textit{Predict}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad [B \to \gamma\bullet](j,k)}{[A \to \alpha B \bullet \beta](i,k)}\ \textit{Complete}$$

Complexity: $\mathcal{O}(|P|^2 n^3)$.

An inference system on items

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \ldots a_n$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ \mathit{Predict}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad [B \to \gamma\bullet](j,k)}{[A \to \alpha B \bullet \beta](i,k)}\ \mathit{Complete}$$

Complexity: $\mathcal{O}(|P|^2 n^3)$.

An inference system on items

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \dots a_n$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ \mathit{Predict}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad [B \to \gamma\bullet](j,k)}{[A \to \alpha B \bullet \beta](i,k)}\ \mathit{Complete}$$

Complexity: $\mathcal{O}(|P|^2 n^3)$.

An inference system on items

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \dots a_n$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ \mathit{Predict}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad [B \to \gamma\bullet](j,k)}{[A \to \alpha B \bullet \beta](i,k)}\ \mathit{Complete}$$

Complexity: $\mathcal{O}(|P|^2 n^3)$.

An inference system on items

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \dots a_n$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \textit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \textit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ \textit{Predict}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad [B \to \gamma\bullet](j,k)}{[A \to \alpha B \bullet \beta](i,k)}\ \textit{Complete}$$

Complexity: $\mathcal{O}(|P|^2 n^3)$.

Making the complexity linear in $|P|$

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \dots a_n$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)} \; Predict$$

$$\Downarrow$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)} \; P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)} \; P2$$

Making the complexity linear in $|P|$

Given a CFG $G = (N, T, P, S)$ and a word $a_1 \dots a_n$

$$\frac{[A \to \alpha \bullet B\beta](i,j) \quad [B \to \gamma\bullet](j,k)}{[A \to \alpha B \bullet \beta](i,k)}\ \textit{Complete}$$

$$\Downarrow$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha B \bullet \beta](i,k)}\ C2$$

A crash introduction to Datalog

Datalog is a recursive query language for relational database.

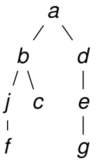
Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$



A crash introduction to Datalog

Datalog is a recursive query language for relational database.

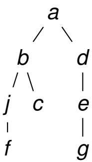
Extensional database

$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database

$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution



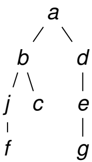
A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

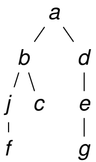
Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution



A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



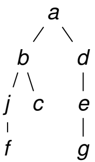
Intentional database
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$sg(a, a)\ sg(b, b)\ sg(c, c)\ sg(d, d)\ sg(e, e)\ sg(j, j)$

A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



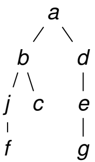
Intentional database
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$sg(a, a)$ $sg(b, b)$ $sg(c, c)$ $sg(d, d)$ $sg(e, e)$ $sg(j, j)$

A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b,j)$
$par(b,c)$
$par(a,b)$
$par(a,d)$
$par(d,e)$
$par(j,f)$
$par(e,g)$



Intentional database
$sg(x,y) :- x = y$
$sg(x,y) :- par(x_p,x),\ sg(x_p,y_p),\ par(y_p,y)$

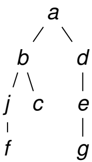
Query: $?-sg(j,x)$
Query resolution: seminaive bottom-up resolution
$sg(a,a)$ $sg(b,b)$ $sg(c,c)$ $sg(d,d)$ $sg(e,e)$ $sg(j,j)$
$sg(j,c)$
$sg(c,j)$

A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database

$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database

$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

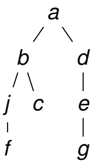
Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$sg(a, a)$ $sg(b, b)$ $sg(c, c)$ $sg(d, d)$ $sg(e, e)$ $sg(j, j)$
$sg(j, c)$
$sg(c, j)$

A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database

$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database

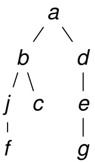
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$sg(a, a)$ $sg(b, b)$ $sg(c, c)$ $sg(d, d)$ $sg(e, e)$ $sg(j, j)$
$sg(j, c)$ $sg(b, d)$
$sg(c, j)$ $sg(d, b)$

A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



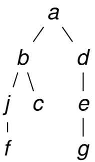
Intentional database
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$sg(a, a)$ $sg(b, b)$ $sg(c, c)$ $sg(d, d)$ $sg(e, e)$ $sg(j, j)$
$sg(j, c)$ $sg(b, d)$
$sg(c, j)$ $sg(d, b)$

A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



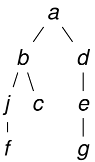
Intentional database
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?{-}sg(j, x)$
Query resolution: seminaive bottom-up resolution
$sg(a, a)$ $sg(b, b)$ $sg(c, c)$ $sg(d, d)$ $sg(e, e)$ $sg(j, j)$
$sg(j, c)$ $sg(b, d)$ $sg(j, e)$
$sg(c, j)$ $sg(d, b)$ $sg(e, j)$

A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$sg(a, a)$ $sg(b, b)$ $sg(c, c)$ $sg(d, d)$ $sg(e, e)$ $sg(j, j)$
$sg(j, c)$ $sg(b, d)$ $sg(j, e)$
$sg(c, j)$ $sg(d, b)$ $sg(e, j)$

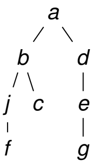
A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

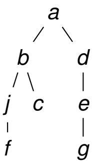
Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$sg(a, a)$ $sg(b, b)$ $sg(c, c)$ $sg(d, d)$ $sg(e, e)$ $sg(j, j)$
$sg(j, c)$ $sg(b, d)$ $sg(j, e)$ $sg(f, g)$
$sg(c, j)$ $sg(d, b)$ $sg(e, j)$ $sg(g, f)$



A crash introduction to Datalog

Datalog is a recursive query language for relational database.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg(x, y) :- x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$sg(a, a)$ $sg(b, b)$ $sg(c, c)$ $sg(d, d)$ $sg(e, e)$ $sg(j, j)$
$sg(j, c)$ $sg(b, d)$ $sg(j, e)$ $sg(f, g)$
$sg(c, j)$ $sg(d, b)$ $sg(e, j)$ $sg(g, f)$
Answer: $\{j; c; e\}$

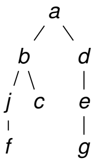
A crash introduction to Datalog: adornment

Profiling the predicates: related to partial evaluation of programs.

Extensional database
$par(b,j)$
$par(b,c)$
$par(a,b)$
$par(a,d)$
$par(d,e)$
$par(j,f)$
$par(e,g)$

Intentional database
$sg\ \ (x,y) :- x = y$
$sg\ \ (x,y) :- par\ \ (x_p, x),\ sg\ \ (x_p, y_p),\ par\ \ (y_p, y)$

Query: $?-sg(j,x)$



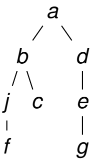
A crash introduction to Datalog: adornment

Profiling the predicates: related to partial evaluation of programs.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg\ \ (x, y) :- x = y$
$sg\ \ (x, y) :- par\ \ (x_p, x),\ sg\ \ (x_p, y_p),\ par\ \ (y_p, y)$

Query: $?-sg(j, x)$
sg^{bf}



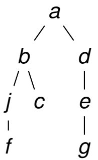
A crash introduction to Datalog: adornment

Profiling the predicates: related to partial evaluation of programs.

Extensional database
$par(b,j)$
$par(b,c)$
$par(a,b)$
$par(a,d)$
$par(d,e)$
$par(j,f)$
$par(e,g)$

Intentional database
$sg^{bf}(x,y) :- x = y$
$sg\ (x,y) :- par\ (x_p,x),\ sg\ (x_p,y_p),\ par\ (y_p,y)$

Query: $?-sg(j,x)$
sg^{bf}



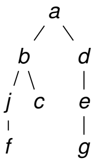
A crash introduction to Datalog: adornment

Profiling the predicates: related to partial evaluation of programs.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg^{bf}(x, y) :- x = y$
$sg^{bf}(x, y) :- par\ (x_p, x),\ sg\ (x_p, y_p),\ par\ (y_p, y)$

Query: $?-sg(j, x)$
sg^{bf}



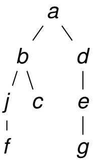
A crash introduction to Datalog: adornment

Profiling the predicates: related to partial evaluation of programs.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg^{bf}(x, y) :- x = y$
$sg^{bf}(x, y) :- par^{fb}(x_p, x),\ sg\ \ (x_p, y_p),\ par\ \ (y_p, y)$

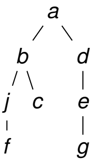
Query: $?- sg(j, x)$
sg^{bf}



A crash introduction to Datalog: adornment

Profiling the predicates: related to partial evaluation of programs.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg^{bf}(x, y) :- x = y$
$sg^{bf}(x, y) :- par^{fb}(x_p, x),\ sg^{bf}(x_p, y_p),\ par\ (y_p, y)$

Query: $?- sg(j, x)$
sg^{bf}

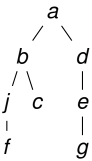
A crash introduction to Datalog: adornment

Profiling the predicates: related to partial evaluation of programs.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg^{bf}(x, y) :- x = y$
$sg^{bf}(x, y) :- par^{fb}(x_p, x),\ sg^{bf}(x_p, y_p),\ par^{bf}(y_p, y)$

Query: $?- sg(j, x)$
sg^{bf}



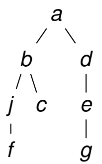
A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg^{bf}(x, y) :- \quad x = y$
$sg^{bf}(x, y) :- par^{fb}(x_p, x),\ sg^{bf}(x_p, y_p),\ par^{bf}(y_p, y)$

Query: $?{-}sg(j, x)$



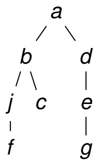
A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- \quad x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?-sg(j, x)$
Magic predicate: $m\_sg(x)$



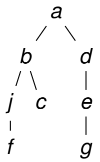
A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x), x = y$
$sg(x, y) :- par(x_p, x), sg(x_p, y_p), par(y_p, y)$

Query: $?-sg(j, x)$
Magic predicate: $m\_sg(x)$



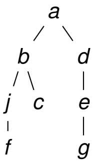
A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$

Query: $?-sg(j, x)$
Magic predicate: $m\_sg(x)$



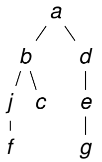
A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- par(x_p, x),\ sg(x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$

Query: $?-sg(j, x)$
Magic predicate: $m\_sg(x)$



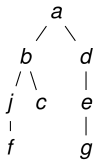
A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$

Query: $?-sg(j, x)$
Magic predicate: $m\_sg(x)$



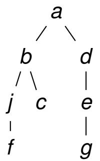
A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$

Query: $?- sg(j, x)$
Magic predicate: $m\_sg(x)$



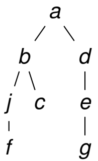
A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$

Query: $?- sg(j, x)$
Magic predicate: $m\_sg(x)$



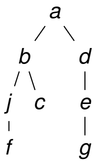
A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$

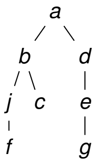
Query: $?-sg(j, x)$
Magic predicate: $m\_sg(x)$



A crash introduction to Datalog: magic set rewriting

Optimization of the intentional database: incremental evaluation of clauses, memoizing intermediate goals.

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- \color{red}{sup_{2,1}(x, x_p)},\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

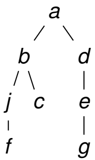
Query: $?-sg(j, x)$
Magic predicate: $m\_sg(x)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$



Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j)$

A crash introduction to Datalog: magic set in action

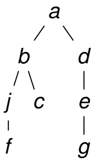
Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database

$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database

$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$



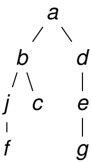
Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database

$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database

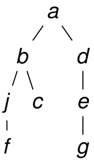
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?{-}sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j)$
$sg(j, j)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

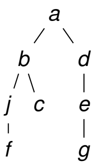
Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j)$
$sg(j, j)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database

$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database

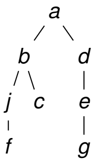
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j)$
$sg(j, j)$
$sup_{2,1}(j, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



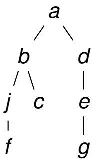
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j)$
$sg(j, j)$
$sup_{2,1}(j, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



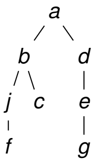
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?{-}sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b)$
$sg(j, j)$
$sup_{2,1}(j, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



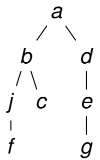
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j)$, $m\_sg(b)$
$sg(j, j)$
$sup_{2,1}(j, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



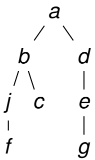
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?{-}sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b)$
$sg(j, j),\ sg(b, b)$
$sup_{2,1}(j, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

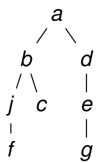
Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b)$
$sg(j, j),\ sg(b, b)$
$sup_{2,1}(j, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database

$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database

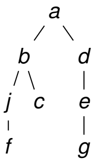
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b)$
$sg(j, j),\ sg(b, b)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



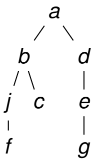
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?{-}sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b)$
$sg(j, j),\ sg(b, b)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



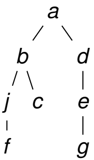
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b)$
$sg(j, j),\ sg(b, b)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



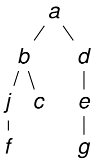
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?{-}sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b)$
$sg(j, j),\ sg(b, b)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b,j)$
$par(b,c)$
$par(a,b)$
$par(a,d)$
$par(d,e)$
$par(j,f)$
$par(e,g)$



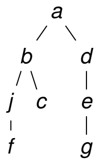
Intentional database
$sg(x,y) :- m\_sg(x),\ x=y$
$sg(x,y) :- sup_{2,2}(x,x_p,y_p),\ par(y_p,y)$
$sup_{2,1}(x,x_p) :- m\_sg(x),\ par(x_p,x)$
$sup_{2,2}(x,x_p,y_p) :- sup_{2,1}(x,x_p),\ sg(x_p,y_p)$
$m\_sg(x_p) :- sup_{2,1}(x,x_p)$

Query: $?-sg(j,x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j,j),\ sg(b,b)$
$sup_{2,1}(j,b),\ sup_{2,1}(b,a)$
$sup_{2,2}(j,b,b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



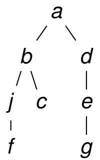
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?{-}sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



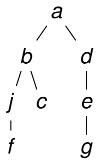
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

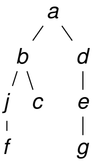
Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database

$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database

$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

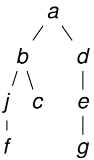
Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c),\ sg(a, a)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b,j)$
$par(b,c)$
$par(a,b)$
$par(a,d)$
$par(d,e)$
$par(j,f)$
$par(e,g)$

Intentional database
$sg(x,y) :- m\_sg(x),\ x = y$
$sg(x,y) :- sup_{2,2}(x,x_p,y_p),\ par(y_p,y)$
$sup_{2,1}(x,x_p) :- m\_sg(x),\ par(x_p,x)$
$sup_{2,2}(x,x_p,y_p) :- sup_{2,1}(x,x_p),\ sg(x_p,y_p)$
$m\_sg(x_p) :- sup_{2,1}(x,x_p)$

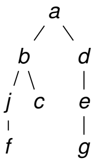


Query: $?-sg(j,x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j,j),\ sg(b,b),\ sg(j,c),\ sg(a,a)$
$sup_{2,1}(j,b),\ sup_{2,1}(b,a)$
$sup_{2,2}(j,b,b)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



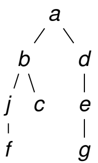
Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c),\ sg(a, a)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b),\ sup_{2,2}(b, a, a)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

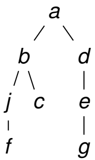
Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c),\ sg(a, a)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b),\ sup_{2,2}(b, a, a)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$



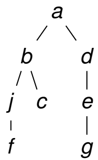
Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c),\ sg(a, a),\ sg(b, d)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b),\ sup_{2,2}(b, a, a)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database

$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database

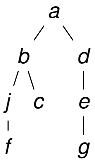
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?-sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c),\ sg(a, a),\ sg(b, d)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b),\ sup_{2,2}(b, a, a)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b,j)$
$par(b,c)$
$par(a,b)$
$par(a,d)$
$par(d,e)$
$par(j,f)$
$par(e,g)$



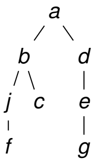
Intentional database
$sg(x,y) :- m\_sg(x),\ x = y$
$sg(x,y) :- sup_{2,2}(x,x_p,y_p),\ par(y_p,y)$
$sup_{2,1}(x,x_p) :- m\_sg(x),\ par(x_p,x)$
$sup_{2,2}(x,x_p,y_p) :- sup_{2,1}(x,x_p),\ sg(x_p,y_p)$
$m\_sg(x_p) :- sup_{2,1}(x,x_p)$

Query: $?-sg(j,x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j,j),\ sg(b,b),\ sg(j,c),\ sg(a,a),\ sg(b,d)$
$sup_{2,1}(j,b),\ sup_{2,1}(b,a)$
$sup_{2,2}(j,b,b),\ sup_{2,2}(b,a,a),\ sup_{2,2}(j,b,d)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

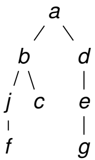
Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c),\ sg(a, a),\ sg(b, d)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b),\ sup_{2,2}(b, a, a),\ sup_{2,2}(j, b, d)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$

Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

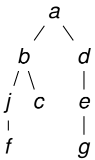


Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c),\ sg(a, a),\ sg(b, d),\ sg(j, e)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b),\ sup_{2,2}(b, a, a),\ sup_{2,2}(j, b, d)$

A crash introduction to Datalog: magic set in action

Magic set + seminaive bottom-up evaluation = SLD-resolution (Selective Linear Definite clause resolution)

Extensional database
$par(b, j)$
$par(b, c)$
$par(a, b)$
$par(a, d)$
$par(d, e)$
$par(j, f)$
$par(e, g)$



Intentional database
$sg(x, y) :- m\_sg(x),\ x = y$
$sg(x, y) :- sup_{2,2}(x, x_p, y_p),\ par(y_p, y)$
$sup_{2,1}(x, x_p) :- m\_sg(x),\ par(x_p, x)$
$sup_{2,2}(x, x_p, y_p) :- sup_{2,1}(x, x_p),\ sg(x_p, y_p)$
$m\_sg(x_p) :- sup_{2,1}(x, x_p)$

Query: $?- sg(j, x)$
Query resolution: seminaive bottom-up resolution
$m\_sg(j),\ m\_sg(b),\ m\_sg(a)$
$sg(j, j),\ sg(b, b),\ sg(j, c),\ sg(a, a),\ sg(b, d),\ sg(j, e)$
$sup_{2,1}(j, b),\ sup_{2,1}(b, a)$
$sup_{2,2}(j, b, b),\ sup_{2,2}(b, a, a),\ sup_{2,2}(j, b, d)$
Answer: $\{j, c, e\}$

Parsing CFGs with Datalog

Given the grammar

$$
\begin{aligned}
S &\rightarrow aSbS \\
S &\rightarrow \epsilon
\end{aligned}
$$

We associate the following *Intentional Database*

$$
\begin{aligned}
S(i,j) &:- \quad a(i,j),\ S(j,k),\ b(k,l),\ S(l,m) \\
S(i,j) &:- \quad i = j
\end{aligned}
$$

Given the word $a_1 \ldots a_n$ we associate the *Extensional Database* $a_1(0,1), \ldots a_i(i-1,i) \ldots a_n(n-1,n)$.
The query $?{-}S(0,n)$ gets a positive answer iff $S \xrightarrow{*} a_1 \ldots a_n$.

Parsing CFGs with Datalog

Given the grammar

$$\begin{aligned} S &\rightarrow aSbS \\ S &\rightarrow \epsilon \end{aligned}$$

We associate the following *Intentional Database*

$$\begin{aligned} S(i,j) &:- \quad a(i,j),\ S(j,k),\ b(k,l),\ S(l,m) \\ S(i,j) &:- \quad i = j \end{aligned}$$

Given the word $a_1 \dots a_n$ we associate the *Extensional Database* $a_1(0,1), \dots a_i(i-1,i) \dots a_n(n-1,n)$.

The query $?{-}S(0,n)$ gets a positive answer iff $S \xrightarrow{*} a_1 \dots a_n$.

Parsing CFGs with Datalog

Given the grammar

$$\begin{array}{rcl} S & \to & aSbS \\ S & \to & \epsilon \end{array}$$

We associate the following *Intentional Database*

$$\begin{array}{rcl} S(i,j) & :- & a(i,j),\ S(j,k),\ b(k,l),\ S(l,m) \\ S(i,j) & :- & i = j \end{array}$$

Given the word $a_1 \dots a_n$ we associate the *Extensional Database* $a_1(0,1), \dots a_i(i-1,i) \dots a_n(n-1,n)$.
The query $?{-}S(0,n)$ gets a positive answer iff $S \xrightarrow{*} a_1 \dots a_n$.

Magic supplementary set rewriting: adornment

Instead of asking the query $?\!-\!S(0,n)$ we may ask for the x's such that $S(0,x)$.

We get the adorned Extensional Database:

$$
\begin{array}{rcl}
S^{bf}(i,m) & :- & a^{bf}(i,j),\ S^{bf}(j,k),\ b^{bf}(k,l),\ S^{bf}(l,m) \\
S^{bf}(i,j) & :- & i=j
\end{array}
$$

We take the magic predicate $m\_S(k)$ and we do the magic set rewriting:

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j
\end{array}
\qquad
\begin{array}{rcl}
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

Magic supplementary set rewriting: adornment

Instead of asking the query $?{-}S(0,n)$ we may ask for the x's such that $S(0,x)$.
We get the adorned Extensional Database:

$$\begin{array}{rcl} S^{bf}(i,m) & :- & a^{bf}(i,j),\ S^{bf}(j,k),\ b^{bf}(k,l),\ S^{bf}(l,m) \\ S^{bf}(i,j) & :- & i=j \end{array}$$

We take the magic predicate $m\_S(k)$ and we do the magic set rewriting:

$$\begin{array}{rcl} sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\ sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\ sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\ S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\ S(i,j) & :- & m\_S(i),\ i=j \end{array}$$

$$\begin{array}{rcl} m\_S(j) & :- & sup_{1,1}(i,j) \\ m\_S(j) & :- & sup_{1,3}(i,j) \end{array}$$

Magic supplementary set rewriting: adornment

Instead of asking the query $?\!-S(0,n)$ we may ask for the x's such that $S(0,x)$.

We get the adorned Extensional Database:

$$
\begin{aligned}
S^{bf}(i,m) &\ :-\ a^{bf}(i,j),\ S^{bf}(j,k),\ b^{bf}(k,l),\ S^{bf}(l,m) \\
S^{bf}(i,j) &\ :-\ i=j
\end{aligned}
$$

We take the magic predicate $m\_S(k)$ and we do the magic set rewriting:

$$
\begin{array}{rcl||rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) & m\_S(j) & :- & sup_{1,1}(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) & m\_S(j) & :- & sup_{1,3}(i,j) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) & & & \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) & & & \\
S(i,j) & :- & m\_S(i),\ i=j & & &
\end{array}
$$

Earley and Datalog

$$\begin{array}{ccc} \multicolumn{3}{c}{a_1 \dots a_n} \\ \hline\hline S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ \textit{Init} \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ \textit{Scan}$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl} sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\ sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\ sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\ S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\ S(i,j) & :- & m\_S(i),\ i=j \\ m\_S(j) & :- & sup_{1,1}(i,j) \\ m\_S(j) & :- & sup_{1,3}(i,j) \\ m\_S(0) & & \end{array}$$

Earley and Datalog

$$\frac{a_1 \dots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet \alpha](0,0)}\ Init \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet \gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma \bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS \bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ Init \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{\dfrac{?S(i)}{[S \rightarrow \bullet aSbS](i,i) \quad a_{i+1} = a}}{[S \rightarrow a \bullet SbS](i,i+1)}$$

Earley and Datalog

$$\frac{a_1 \dots a_n}{\begin{array}{lcl} S & \to & aSbS \\ S & \to & \epsilon \end{array}}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ Init \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i = j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$[S \to \bullet aSbS](i,j)$
$[S \to a \bullet SbS](i,j)$
$[S \to aS \bullet bS](i,j)$
$[S \to aSb \bullet S](i,j)$
$[S \to aSbS\bullet](i,j)$
$[S \to \bullet](i,j)$
$S(i,j)$
$?S(i)$

$$\frac{\dfrac{?S(i)}{[S \to \bullet aSbS](i,i) \quad a_{i+1} = a}}{[S \to a \bullet SbS](i,i+1)}$$

$$\frac{m\_S(i) \quad a(i,i+1)}{sup_{1,1}(i,i+1)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \to & aSbS \\ S & \to & \epsilon \end{array}}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ Init \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \to \bullet aSbS](i,j) \\
[S \to a \bullet SbS](i,j) \\
[S \to aS \bullet bS](i,j) \\
[S \to aSb \bullet S](i,j) \\
[S \to aSbS\bullet](i,j) \\
[S \to \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{\dfrac{?S(i)}{[S \to \bullet aSbS](i,i) \quad a_{i+1} = a}}{[S \to a \bullet SbS](i,i+1)}$$

$$\frac{m\_S(i) \quad a(i,i+1)}{sup_{1,1}(i,i+1)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lll} S & \to & aSbS \\ S & \to & \epsilon \end{array}}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \to \bullet aSbS](i,j) \\
[S \to a \bullet SbS](i,j) \\
[S \to aS \bullet bS](i,j) \\
[S \to aSb \bullet S](i,j) \\
[S \to aSbS\bullet](i,j) \\
[S \to \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{[S \to a \bullet SbS](i,j) \quad S(j,k)}{[S \to aS \bullet bS](i,k)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \dots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ Init \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{[S \rightarrow a \bullet SbS](i,j) \quad S(j,k)}{[S \rightarrow aS \bullet bS](i,k)}$$

$$\frac{sup_{1,1}(i,j) \quad S(j,k)}{sup_{1,2}(i,k)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j)
\end{array}$$

Earley and Datalog

$$\begin{array}{c} a_1 \dots a_n \\ \hline\hline \begin{array}{ccc} S & \to & aSbS \\ S & \to & \epsilon \end{array} \end{array}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl} sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\ sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\ sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\ S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\ S(i,j) & :- & m\_S(i),\ i=j \\ m\_S(j) & :- & sup_{1,1}(i,j) \\ m\_S(j) & :- & sup_{1,3}(i,j) \\ m\_S(0) & & \end{array}$$

$$\begin{array}{c} [S \to \bullet aSbS](i,j) \\ [S \to a \bullet SbS](i,j) \\ [S \to aS \bullet bS](i,j) \\ [S \to aSb \bullet S](i,j) \\ [S \to aSbS\bullet](i,j) \\ [S \to \bullet](i,j) \\ S(i,j) \\ ?S(i) \end{array}$$

$$\frac{[S \to a \bullet SbS](i,j) \quad S(j,k)}{[S \to aS \bullet bS](i,k)}$$

$$\frac{sup_{1,1}(i,j) \quad S(j,k)}{sup_{1,2}(i,k)}$$

$$\begin{array}{rcl} ?S(i) & \approx & m\_S(i) \\ a_{i+1} = a & \approx & a(i,i+1) \\ [S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\ [S \to aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\ S(i,j) & \approx & S(i,j) \end{array}$$

Earley and Datalog

$$\frac{a_1 \dots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl} sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\ sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\ sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\ S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\ S(i,j) & :- & m\_S(i),\ i=j \\ m\_S(j) & :- & sup_{1,1}(i,j) \\ m\_S(j) & :- & sup_{1,3}(i,j) \\ m\_S(0) & & \end{array}$$

$$\begin{array}{c} [S \rightarrow \bullet aSbS](i,j) \\ [S \rightarrow a \bullet SbS](i,j) \\ [S \rightarrow aS \bullet bS](i,j) \\ [S \rightarrow aSb \bullet S](i,j) \\ [S \rightarrow aSbS\bullet](i,j) \\ [S \rightarrow \bullet](i,j) \\ S(i,j) \\ ?S(i) \end{array}$$

$$\frac{[S \rightarrow a \bullet SbS](i,j)}{?S(j)}$$

$$\begin{array}{rcl} ?S(i) & \approx & m\_S(i) \\ a_{i+1} = a & \approx & a(i,i+1) \\ [S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\ [S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\ S(i,j) & \approx & S(i,j) \end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \to & aSbS \\ S & \to & \epsilon \end{array}}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ Init \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \to \bullet aSbS](i,j) \\
[S \to a \bullet SbS](i,j) \\
[S \to aS \bullet bS](i,j) \\
[S \to aSb \bullet S](i,j) \\
[S \to aSbS\bullet](i,j) \\
[S \to \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{[S \to a \bullet SbS](i,j)}{?S(j)} \qquad \frac{sup_{1,2}(i,j)}{m\_S(j)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \to aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \dots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ Init \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i = j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{[S \rightarrow a \bullet SbS](i,j)}{?S(j)} \qquad \frac{sup_{1,2}(i,j)}{m\_S(j)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ \textit{Init} \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ \textit{Scan}$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{[S \rightarrow aS \bullet bS](i,j) \quad a_{j+1} = b}{[S \rightarrow aSb \bullet S](i,j)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j)
\end{array}$$

Earley and Datalog

$$\begin{array}{l} a_1 \ldots a_n \\ \hline\hline S \rightarrow aSbS \\ S \rightarrow \epsilon \end{array}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ Init \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl} sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\ sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\ sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\ S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\ S(i,j) & :- & m\_S(i),\ i=j \\ m\_S(j) & :- & sup_{1,1}(i,j) \\ m\_S(j) & :- & sup_{1,3}(i,j) \\ m\_S(0) & & \end{array}$$

$$\begin{array}{c} [S \rightarrow \bullet aSbS](i,j) \\ [S \rightarrow a \bullet SbS](i,j) \\ [S \rightarrow aS \bullet bS](i,j) \\ [S \rightarrow aSb \bullet S](i,j) \\ [S \rightarrow aSbS\bullet](i,j) \\ [S \rightarrow \bullet](i,j) \\ S(i,j) \\ ?S(i) \end{array}$$

$$\frac{[S \rightarrow aS \bullet bS](i,j) \quad a_{j+1} = b}{[S \rightarrow aSb \bullet S](i,j)}$$

$$\frac{sup_{1,2}(i,j) \quad b(j,j+1)}{sup_{1,3}(i,j+1)}$$

$$\begin{array}{rcl} ?S(i) & \approx & m\_S(i) \\ a_{i+1} = a & \approx & a(i,i+1) \\ [S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\ [S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\ S(i,j) & \approx & S(i,j) \\ [S \rightarrow aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j) \end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ Init \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{[S \rightarrow aS \bullet bS](i,j) \quad a_{j+1} = b}{[S \rightarrow aSb \bullet S](i,j)}$$

$$\frac{sup_{1,2}(i,j) \quad b(j,j+1)}{sup_{1,3}(i,j+1)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j) \\
[S \rightarrow aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \to & aSbS \\ S & \to & \epsilon \end{array}}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl} sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\ sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\ sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\ S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\ S(i,j) & :- & m\_S(i),\ i=j \\ m\_S(j) & :- & sup_{1,1}(i,j) \\ m\_S(j) & :- & sup_{1,3}(i,j) \\ m\_S(0) & & \end{array}$$

$$\begin{array}{c} [S \to \bullet aSbS](i,j) \\ [S \to a \bullet SbS](i,j) \\ [S \to aS \bullet bS](i,j) \\ [S \to aSb \bullet S](i,j) \\ [S \to aSbS\bullet](i,j) \\ [S \to \bullet](i,j) \\ S(i,j) \\ ?S(i) \end{array}$$

$$\cfrac{\cfrac{[S \to aSb \bullet S](i,j) \quad S(j,k)}{[S \to aSbS\bullet](i,k)}}{S(i,k)}$$

$$\begin{array}{rcl} ?S(i) & \approx & m\_S(i) \\ a_{i+1} = a & \approx & a(i,i+1) \\ [S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\ [S \to aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\ S(i,j) & \approx & S(i,j) \\ [S \to aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j) \end{array}$$

Earley and Datalog

$$\frac{a_1 \dots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ Init \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\cfrac{\cfrac{[S \rightarrow aSb \bullet S](i,j) \quad S(j,k)}{[S \rightarrow aSbS\bullet](i,k)}}{S(i,k)}$$

$$\frac{sup_{2,3}(i,j) \quad S(j,k)}{S(j,k)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j) \\
[S \rightarrow aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \dots a_n}{\begin{array}{lcl} S & \to & aSbS \\ S & \to & \epsilon \end{array}}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i = j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \to \bullet aSbS](i,j) \\
[S \to a \bullet SbS](i,j) \\
[S \to aS \bullet bS](i,j) \\
[S \to aSb \bullet S](i,j) \\
[S \to aSbS\bullet](i,j) \\
[S \to \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{\dfrac{[S \to aSb \bullet S](i,j) \quad S(j,k)}{[S \to aSbS\bullet](i,k)}}{S(i,k)} \qquad \frac{sup_{2,3}(i,j) \quad S(j,k)}{S(j,k)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \to aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j) \\
[S \to aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \to & aSbS \\ S & \to & \epsilon \end{array}}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ Init \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \to \bullet aSbS](i,j) \\
[S \to a \bullet SbS](i,j) \\
[S \to aS \bullet bS](i,j) \\
[S \to aSb \bullet S](i,j) \\
[S \to aSbS\bullet](i,j) \\
[S \to \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{[S \to aSb \bullet S](i,j)}{?S(j)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \to aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j) \\
[S \to aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet \alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{[S \rightarrow aSb \bullet S](i,j)}{?S(j)} \qquad \frac{sup_{1,3}(i,j)}{m\_S(j)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j) \\
[S \rightarrow aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j)
\end{array}$$

Earley and Datalog

$$\begin{array}{ccc} a_1 \dots a_n & & \\ \hline\hline S & \to & aSbS \\ S & \to & \epsilon \end{array}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl} sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\ sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\ sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\ S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\ S(i,j) & :- & m\_S(i),\ i=j \\ m\_S(j) & :- & sup_{1,1}(i,j) \\ m\_S(j) & :- & sup_{1,3}(i,j) \\ m\_S(0) & & \end{array}$$

$$\begin{array}{c} [S \to \bullet aSbS](i,j) \\ [S \to a \bullet SbS](i,j) \\ [S \to aS \bullet bS](i,j) \\ [S \to aSb \bullet S](i,j) \\ [S \to aSbS\bullet](i,j) \\ [S \to \bullet](i,j) \\ S(i,j) \\ ?S(i) \end{array}$$

$$\frac{[S \to aSb \bullet S](i,j)}{?S(j)} \qquad \frac{sup_{1,3}(i,j)}{m\_S(j)}$$

$$\begin{array}{rcl} ?S(i) & \approx & m\_S(i) \\ a_{i+1} = a & \approx & a(i,i+1) \\ [S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\ [S \to aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\ S(i,j) & \approx & S(i,j) \\ [S \to aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j) \end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{\dfrac{?S(i)}{[S \rightarrow \bullet](i,i)}}{S(i,i)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j) \\
[S \rightarrow aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \ldots a_n}{\begin{array}{lcl} S & \to & aSbS \\ S & \to & \epsilon \end{array}}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ \mathit{Init} \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ \mathit{Scan}$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \to \bullet aSbS](i,j) \\
[S \to a \bullet SbS](i,j) \\
[S \to aS \bullet bS](i,j) \\
[S \to aSb \bullet S](i,j) \\
[S \to aSbS\bullet](i,j) \\
[S \to \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{\dfrac{?S(i)}{[S \to \bullet](i,i)}}{S(i,i)} \qquad \frac{m\_S(i) \quad i=j}{S(i,j)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \to aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j) \\
[S \to aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \dots a_n}{\begin{array}{lcl} S & \rightarrow & aSbS \\ S & \rightarrow & \epsilon \end{array}}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ Init \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{\dfrac{?S(i)}{[S \rightarrow \bullet](i,i)}}{S(i,i)} \qquad \frac{m\_S(i) \quad i=j}{S(i,j)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \rightarrow a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \rightarrow aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j) \\
[S \rightarrow aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j)
\end{array}$$

Earley and Datalog

$$\frac{a_1 \dots a_n}{\begin{array}{lcl} S & \to & aSbS \\ S & \to & \epsilon \end{array}}$$

$$\frac{S \to \alpha \in P}{[S \to \bullet\alpha](0,0)}\ Init \qquad \frac{[A \to \alpha \bullet a_{j+1}\beta](i,j)}{[A \to \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \to \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \to \gamma \in P}{[B \to \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \to \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \to \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \to \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i = j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \to \bullet aSbS](i,j) \\
[S \to a \bullet SbS](i,j) \\
[S \to aS \bullet bS](i,j) \\
[S \to aSb \bullet S](i,j) \\
[S \to aSbS\bullet](i,j) \\
[S \to \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

$$\frac{?S(0)}{[S \to \bullet aSbS](0,0)} \qquad \frac{\dfrac{?S(0)}{[S \to \bullet](0,0)}}{S(0,0)}$$

$$\begin{array}{rcl}
?S(i) & \approx & m_S(i) \\
a_{i+1} = a & \approx & a(i,i+1) \\
[S \to a \bullet SbS](i,j) & \approx & sup_{1,1}(i,j) \\
[S \to aS \bullet bS](i,j) & \approx & sup_{1,2}(i,j) \\
S(i,j) & \approx & S(i,j) \\
[S \to aSb \bullet S](i,j) & \approx & sup_{1,3}(i,j)
\end{array}$$

Earley and Datalog

$$\begin{array}{l} a_1 \ldots a_n \\ \hline\hline S \rightarrow aSbS \\ S \rightarrow \epsilon \end{array}$$

$$\frac{S \rightarrow \alpha \in P}{[S \rightarrow \bullet\alpha](0,0)}\ Init \qquad \frac{[A \rightarrow \alpha \bullet a_{j+1}\beta](i,j)}{[A \rightarrow \alpha a_{j+1} \bullet \beta](i,j+1)}\ Scan$$

$$\frac{[A \rightarrow \alpha \bullet B\beta](i,j)}{?B(j)}\ P1 \qquad \frac{?B(j) \quad B \rightarrow \gamma \in P}{[B \rightarrow \bullet\gamma](j,j)}\ P2$$

$$\frac{[B \rightarrow \gamma\bullet](j,k)}{B(j,k)}\ C1 \qquad \frac{[A \rightarrow \alpha \bullet B\beta](i,j) \quad B(j,k)}{[A \rightarrow \alpha \bullet \beta](i,k)}\ C2$$

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j) \\
m_S(0) & &
\end{array}$$

$$\begin{array}{c}
[S \rightarrow \bullet aSbS](i,j) \\
[S \rightarrow a \bullet SbS](i,j) \\
[S \rightarrow aS \bullet bS](i,j) \\
[S \rightarrow aSb \bullet S](i,j) \\
[S \rightarrow aSbS\bullet](i,j) \\
[S \rightarrow \bullet](i,j) \\
S(i,j) \\
?S(i)
\end{array}$$

EARLEY = DATALOG + MAGIC SET

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

An example

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j)
\end{array}$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

An example

$$\begin{array}{rcl}
\color{red}{sup_{1,1}(i,j)} & \color{red}{:-} & \color{red}{m_S(i),\ a(i,j)} \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j)
\end{array}$$

ababba

$\color{red}{a(0,1)}$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$\color{red}{m\_S(0)}$

An example

$$
\begin{array}{rcl}
\color{red}{sup_{1,1}(i,j)} & \color{red}{:-} & \color{red}{m\_S(i),\ a(i,j)} \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$\color{red}{a(0,1)}$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$\color{red}{m\_S(0)}$

$sup_{1,1}(0,1)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i = j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $S(0,0)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $S(0,0)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $m\_S(1)$ $S(0,0)$

An example

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j)
\end{array}$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $m\_S(1)$ $S(0,0)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $m\_S(1)$ $S(0,0)$
$S(1,1)$

An example

$$\begin{array}{rcl} sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\ \color{red}{sup_{1,2}(i,k)} & \color{red}{:-} & \color{red}{sup_{1,1}(i,j),\ S(j,k)} \\ sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\ S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\ S(i,j) & :- & m\_S(i),\ i=j \\ m\_S(j) & :- & sup_{1,1}(i,j) \\ m\_S(j) & :- & sup_{1,3}(i,j) \end{array}$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$\color{red}{sup_{1,1}(0,1)}$ $m\_S(1)$ $S(0,0)$
$\color{red}{S(1,1)}$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $sup_{1,2}(0,1)$ $m\_S(1)$ $S(0,0)$
$S(1,1)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $sup_{1,2}(0,1)$ $m\_S(1)$ $S(0,0)$
$S(1,1)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
				$S(1,1)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
\color{red}{m\_S(j)} & \color{red}{:-} & \color{red}{sup_{1,3}(i,j)}
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$\color{red}{sup_{1,3}(0,2)}$	$m\_S(1)$	$S(0,0)$
				$S(1,1)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $sup_{1,2}(0,1)$ $sup_{1,3}(0,2)$ $m\_S(1)$ $S(0,0)$
$m\_S(2)$ $S(1,1)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $sup_{1,2}(0,1)$ $sup_{1,3}(0,2)$ $m\_S(1)$ $S(0,0)$
$m\_S(2)$ $S(1,1)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $sup_{1,2}(0,1)$ $sup_{1,3}(0,2)$

$m\_S(1)$
$m\_S(2)$

$S(0,0)$
$S(1,1)$
$S(2,2)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$ $sup_{1,2}(0,1)$ $sup_{1,3}(0,2)$

$m\_S(1)$ $m\_S(2)$

$S(0,0)$ $S(1,1)$ $S(2,2)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
			$m\_S(2)$	$S(1,1)$
				$S(2,2)$
				$S(0,2)$

An example

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j)
\end{array}$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
			$m\_S(2)$	$S(1,1)$
				$S(2,2)$
				$S(0,2)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$$
\begin{array}{llllll}
sup_{1,1}(0,1) & sup_{1,2}(0,1) & sup_{1,3}(0,2) & & m\_S(1) & S(0,0) \\
sup_{1,1}(2,3) & & & & m\_S(2) & S(1,1) \\
 & & & & & S(2,2) \\
 & & & & & S(0,2)
\end{array}
$$

An example

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i = j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j)
\end{array}$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$			$m\_S(2)$	$S(1,1)$
				$S(2,2)$
				$S(0,2)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$$
\begin{array}{l}
a(0,1) \\
b(1,2) \\
a(2,3) \\
b(3,4) \\
b(4,5) \\
a(5,6) \\
m\_S(0)
\end{array}
$$

$$
\begin{array}{llllll}
sup_{1,1}(0,1) & sup_{1,2}(0,1) & sup_{1,3}(0,2) & & m\_S(1) & S(0,0) \\
sup_{1,1}(2,3) & & & & m\_S(2) & S(1,1) \\
 & & & & m\_S(3) & S(2,2) \\
 & & & & & S(0,2)
\end{array}
$$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$			$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
				$S(0,2)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$			$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
				$S(0,2)$
				$S(3,3)$

An example

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j)
\end{array}$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$			$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
				$S(0,2)$
				$S(3,3)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$		$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
				$S(0,2)$
				$S(3,3)$

An example

$$\begin{array}{rcl} sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\ sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\ sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\ S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\ S(i,j) & :- & m\_S(i),\ i=j \\ m\_S(j) & :- & sup_{1,1}(i,j) \\ m\_S(j) & :- & sup_{1,3}(i,j) \end{array}$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$		$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
				$S(0,2)$
				$S(3,3)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$	$sup_{1,3}(2,4)$	$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
				$S(0,2)$
				$S(3,3)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$$
\begin{array}{l}
a(0,1) \\
b(1,2) \\
a(2,3) \\
b(3,4) \\
b(4,5) \\
a(5,6) \\
m\_S(0)
\end{array}
$$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$	$sup_{1,3}(2,4)$	$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
				$S(0,2)$
				$S(3,3)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$	$sup_{1,3}(2,4)$	$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
			$m\_S(4)$	$S(0,2)$
				$S(3,3)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$$
\begin{array}{lllll}
sup_{1,1}(0,1) & sup_{1,2}(0,1) & sup_{1,3}(0,2) & m\_S(1) & S(0,0) \\
sup_{1,1}(2,3) & sup_{1,2}(2,3) & sup_{1,3}(2,4) & m\_S(2) & S(1,1) \\
 & & & m\_S(3) & S(2,2) \\
 & & & m\_S(4) & S(0,2) \\
 & & & & S(3,3)
\end{array}
$$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$	$sup_{1,3}(2,4)$	$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
			$m\_S(4)$	$S(0,2)$
				$S(3,3)$
				$S(4,4)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$	$sup_{1,3}(2,4)$	$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
			$m\_S(4)$	$S(0,2)$
				$S(3,3)$
				$S(4,4)$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$	$sup_{1,3}(2,4)$	$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
			$m\_S(4)$	$S(0,2)$
				$S(3,3)$
				$S(4,4)$
				$S(2,4)$

An example

$$\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
\color{red}{S(i,j)} & \color{red}{:-} & \color{red}{sup_{1,3}(i,l),\ S(l,m)} \\
S(i,j) & :- & m_S(i),\ i=j \\
m_S(j) & :- & sup_{1,1}(i,j) \\
m_S(j) & :- & sup_{1,3}(i,j)
\end{array}$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$\color{red}{sup_{1,3}(0,2)}$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$	$sup_{1,3}(2,4)$	$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
			$m\_S(4)$	$S(0,2)$
				$S(3,3)$
				$S(4,4)$
				$\color{red}{S(2,4)}$

An example

$$
\begin{array}{rcl}
sup_{1,1}(i,j) & :- & m\_S(i),\ a(i,j) \\
sup_{1,2}(i,k) & :- & sup_{1,1}(i,j),\ S(j,k) \\
sup_{1,3}(i,l) & :- & sup_{1,2}(i,k),\ b(k,l) \\
S(i,j) & :- & sup_{1,3}(i,l),\ S(l,m) \\
S(i,j) & :- & m\_S(i),\ i=j \\
m\_S(j) & :- & sup_{1,1}(i,j) \\
m\_S(j) & :- & sup_{1,3}(i,j)
\end{array}
$$

ababba

$a(0,1)$
$b(1,2)$
$a(2,3)$
$b(3,4)$
$b(4,5)$
$a(5,6)$
$m\_S(0)$

$sup_{1,1}(0,1)$	$sup_{1,2}(0,1)$	$sup_{1,3}(0,2)$	$m\_S(1)$	$S(0,0)$
$sup_{1,1}(2,3)$	$sup_{1,2}(2,3)$	$sup_{1,3}(2,4)$	$m\_S(2)$	$S(1,1)$
			$m\_S(3)$	$S(2,2)$
			$m\_S(4)$	$S(0,2)$
				$S(3,3)$
				$S(4,4)$
				$S(2,4)$
				$S(0,4)$

Outline

Datalog technique for parsing

- Kanazawa (2008)

A Prefix-Correct Earley Recognizer for Multiple Context-Free Grammars

Since it involves very little non-standard technique, we believe that our method is easier to understand and easier to prove correct than previous approaches. For this reason, we also hope that this work serves useful pedagogical purposes.

From MCFGs to Datalog: an example by Kanazawa

$$\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}$$

$$S({}_i\, x_1\, {}_j\, y_1\, {}_k\, x_2\, {}_l\, y_2\, {}_m) \leftarrow P({}_i\, x_1\, {}_j,\ {}_k\, x_2\, {}_l),\ Q({}_j\, y_1\, {}_k,\ {}_l\, y_2\, {}_m)$$

From MCFGs to Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$S({}_i\, x_1\, {}_j\, y_1\, {}_k\, x_2\, {}_l\, y_2\, {}_m) \leftarrow P({}_i\, x_1\, {}_j,\ {}_k\, x_2\, {}_l),\ Q({}_j\, y_1\, {}_k,\ {}_l\, y_2\, {}_m)$$

$$S(i, m) :- P(i, j, k, l),\ Q(j, k, l, m)$$

From MCFGs to Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
P({}_i\, a_1\, {}_j\, x_1\, {}_k\, a_2\, {}_l,\ {}_m\, a_3\, {}_n\, x_2\, {}_o\, a_4\, {}_p) \leftarrow P({}_j\, x_1\, {}_k,\ {}_n\, x_2\, {}_o)
$$

From MCFGs to Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$P({}_i\ a_1\ {}_j\ x_1\ {}_k\ a_2\ {}_l,\ {}_m\ a_3\ {}_n\ x_2\ {}_o\ a_4\ {}_p) \leftarrow P({}_j\ x_1\ {}_k,\ {}_n\ x_2\ {}_o)$$

$$P(i, l, m, p) :- a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p)$$

From MCFGs to Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
S^{bf}(i, m) & :- & P^{bfff}(i, j, k, l),\ Q^{bbbf}(j, k, l, m) \\
P^{bfff}(i, k, l, n) & :- & a_1^{bf}(i, j),\ a_2^{bf}(j, k),\ a_3^{ff}(l, m),\ a_4^{bf}(m, n) \\
P^{bfff}(i, l, m, p) & :- & a_1^{bf}(i, j),\ P^{bfff}(j, k, n, o),\ a_2^{bf}(k, l),\ a_3^{fb}(m, n),\ a_4^{bf}(o, p) \\
Q^{bbbf}(i, k, l, n) & :- & a_1^{bf}(i, j),\ b_2^{bb}(j, k),\ b_3^{bf}(l, m),\ b_4^{bb}(m, n) \\
Q^{bbbf}(i, l, m, p) & :- & b_1^{bf}(i, j),\ b_2^{fb}(k, l),\ b_3^{fb}(m, n),\ Q^{bbbf}(j, k, n, o),\ b_4^{bf}(o, p)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
S^{bf}(i, m) & :- & P^{bfff}(i, j, k, l),\ Q^{bbbf}(j, k, l, m) \\
P^{bfff}(i, k, l, n) & :- & a_1^{bf}(i, j),\ a_2^{bf}(j, k),\ a_3^{ff}(l, m),\ a_4^{bf}(m, n) \\
P^{bfff}(i, l, m, p) & :- & a_1^{bf}(i, j),\ P^{bfff}(j, k, n, o),\ a_2^{bf}(k, l),\ a_3^{fb}(m, n),\ a_4^{bf}(o, p) \\
Q^{bbbf}(i, k, l, n) & :- & a_1^{bf}(i, j),\ b_2^{bb}(j, k),\ b_3^{bf}(l, m),\ b_4^{bb}(m, n) \\
Q^{bbbf}(i, l, m, p) & :- & b_1^{bf}(i, j),\ b_2^{fb}(k, l),\ b_3^{fb}(m, n),\ Q^{bbbf}(j, k, n, o),\ b_4^{bf}(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
m\_P(i) & :- & m\_S(i) \\
m\_Q(j, k, l) & :- & sup_{1,1}(i, j, k, l) \\
m\_P(j) & :- & sup_{3,1}(i, j) \\
m\_Q(j, k, n) & :- & sup_{5,3}(i, j, k, l, n) \\
 & & \\
sup_{1,1}(i, j, k, l) & :- & m\_S(i),\ P(i, j, k, l) \\
sup_{2,1}(i, j) & :- & m\_P(i),\ a_1(i, j) \\
sup_{2,2}(i, k) & :- & sup_{2,1}(i, j),\ a_2(j, k) \\
sup_{2,3}(i, k, l, m) & :- & sup_{2,2}(i, k)\ a_3(l, m) \\
sup_{3,1}(i, j) & :- & m\_P(i),\ a_1(i, j) \\
sup_{3,2}(i, k, n, o) & :- & sup_{3,1}(i, j),\ P(j, k, n, o) \\
sup_{3,3}(i, l, n, o) & :- & sup_{3,2}(i, k, n, o),\ a_2(k, l) \\
sup_{3,4}(i, l, m, o) & :- & sup_{3,3}(i, l, n, o),\ a_3(m, n) \\
sup_{4,1}(i, j, k, l) & :- & m\_Q(i, k, l),\ b_1(i, j) \\
sup_{4,2}(i, k, l) & :- & sup_{4,1}(i, j, k, l),\ b_2(j, k) \\
sup_{4,3}(i, k, l, m) & :- & sup_{4,2}(i, k, l),\ b_3(l, m) \\
sup_{5,1}(i, j, l, m) & :- & m\_Q(i, l, m),\ b_1(i, j) \\
sup_{5,2}(i, j, k, l, m) & :- & sup_{5,1}(i, j, l, m),\ b_2(k, l) \\
sup_{5,3}(i, j, k, l, m, n) & :- & sup_{5,2}(i, j, k, l, m),\ b_3(m, n) \\
sup_{5,4}(i, l, m, o) & :- & sup_{5,3}(i, j, k, l, m, n),\ Q(j, k, n, o) \\
 & & \\
S(i, m) & :- & sup_{1,1}(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & sup_{2,3}(i, k, l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & sup_{3,4}(i, l, m, o),\ a_4(o, p) \\
Q(i, k, l, n) & :- & sup_{4,3}(i, k, l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & sup_{5,4}(i, l, m, o),\ b_4(o, p)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
m\_P(i) & :- & m\_S(i) \\
m\_Q(j,k,l) & :- & sup_{1,1}(i,j,k,l) \\
m\_P(j) & :- & sup_{3,1}(i,j) \\
m\_Q(j,k,n) & :- & sup_{5,3}(i,j,k,l,n) \\
\\
sup_{1,1}(i,j,k,l) & :- & m\_S(i),\ P(i,j,k,l) \\
sup_{2,1}(i,j) & :- & m\_P(i),\ a_1(i,j) \\
sup_{2,2}(i,k) & :- & sup_{2,1}(i,j),\ a_2(j,k) \\
sup_{2,3}(i,k,l,m) & :- & sup_{2,2}(i,k)\ a_3(l,m) \\
sup_{3,1}(i,j) & :- & m\_P(i),\ a_1(i,j) \\
sup_{3,2}(i,k,n,o) & :- & sup_{3,1}(i,j),\ P(j,k,n,o) \\
sup_{3,3}(i,l,n,o) & :- & sup_{3,2}(i,k,n,o),\ a_2(k,l) \\
sup_{3,4}(i,l,m,o) & :- & sup_{3,3}(i,l,n,o),\ a_3(m,n) \\
sup_{4,1}(i,j,k,l) & :- & m\_Q(i,k,l),\ b_1(i,j) \\
sup_{4,2}(i,k,l) & :- & sup_{4,1}(i,j,k,l),\ b_2(j,k) \\
sup_{4,3}(i,k,l,m) & :- & sup_{4,2}(i,k,l),\ b_3(l,m) \\
sup_{5,1}(i,j,l,m) & :- & m\_Q(i,l,m),\ b_1(i,j) \\
sup_{5,2}(i,j,k,l,m) & :- & sup_{5,1}(i,j,l,m),\ b_2(k,l) \\
sup_{5,3}(i,j,k,l,m,n) & :- & sup_{5,2}(i,j,k,l,m),\ b_3(m,n) \\
sup_{5,4}(i,l,m,o) & :- & sup_{5,3}(i,j,k,l,m,n),\ Q(j,k,n,o) \\
\\
S(i,m) & :- & sup_{1,1}(i,j,k,l),\ Q(j,k,l,m) \\
P(i,k,l,n) & :- & sup_{2,3}(i,k,l,m),\ a_4(m,n) \\
P(i,l,m,p) & :- & sup_{3,4}(i,l,m,o),\ a_4(o,p) \\
Q(i,k,l,n) & :- & sup_{4,3}(i,k,l,m),\ b_4(m,n) \\
Q(i,l,m,p) & :- & sup_{5,4}(i,l,m,o),\ b_4(o,p)
\end{array}
$$

$a_1 a_2 a_3 a_4$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
m\_P(i) & :- & m\_S(i) \\
m\_Q(j,k,l) & :- & sup_{1,1}(i,j,k,l) \\
m\_P(j) & :- & sup_{3,1}(i,j) \\
m\_Q(j,k,n) & :- & sup_{5,3}(i,j,k,l,n) \\
\\
sup_{1,1}(i,j,k,l) & :- & m\_S(i),\ P(i,j,k,l) \\
sup_{2,1}(i,j) & :- & m\_P(i),\ a_1(i,j) \\
sup_{2,2}(i,k) & :- & sup_{2,1}(i,j),\ a_2(j,k) \\
sup_{2,3}(i,k,l,m) & :- & sup_{2,2}(i,k)\ a_3(l,m) \\
sup_{3,1}(i,j) & :- & m\_P(i),\ a_1(i,j) \\
sup_{3,2}(i,k,n,o) & :- & sup_{3,1}(i,j),\ P(j,k,n,o) \\
sup_{3,3}(i,l,n,o) & :- & sup_{3,2}(i,k,n,o),\ a_2(k,l) \\
sup_{3,4}(i,l,m,o) & :- & sup_{3,3}(i,l,n,o),\ a_3(m,n) \\
sup_{4,1}(i,j,k,l) & :- & m\_Q(i,k,l),\ b_1(i,j) \\
sup_{4,2}(i,k,l) & :- & sup_{4,1}(i,j,k,l),\ b_2(j,k) \\
sup_{4,3}(i,k,l,m) & :- & sup_{4,2}(i,k,l),\ b_3(l,m) \\
sup_{5,1}(i,j,l,m) & :- & m\_Q(i,l,m),\ b_1(i,j) \\
sup_{5,2}(i,j,k,l,m) & :- & sup_{5,1}(i,j,l,m),\ b_2(k,l) \\
sup_{5,3}(i,j,k,l,m,n) & :- & sup_{5,2}(i,j,k,l,m),\ b_3(m,n) \\
sup_{5,4}(i,l,m,o) & :- & sup_{5,3}(i,j,k,l,m,n),\ Q(j,k,n,o) \\
\\
S(i,m) & :- & sup_{1,1}(i,j,k,l),\ Q(j,k,l,m) \\
P(i,k,l,n) & :- & sup_{2,3}(i,k,l,m),\ a_4(m,n) \\
P(i,l,m,p) & :- & sup_{3,4}(i,l,m,o),\ a_4(o,p) \\
Q(i,k,l,n) & :- & sup_{4,3}(i,k,l,m),\ b_4(m,n) \\
Q(i,l,m,p) & :- & sup_{5,4}(i,l,m,o),\ b_4(o,p)
\end{array}
$$

$a_1 a_2 a_3 a_4$

$$
\begin{array}{c}
a_1(0,1) \\
a_2(1,2) \\
a_3(2,3) \\
a_4(3,4) \\
m\_S(0)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
m\_P(i) & :- & m\_S(i) \\
m\_Q(j,k,l) & :- & sup_{1,1}(i,j,k,l) \\
m\_P(j) & :- & sup_{3,1}(i,j) \\
m\_Q(j,k,n) & :- & sup_{5,3}(i,j,k,l,n) \\
\\
sup_{1,1}(i,j,k,l) & :- & m\_S(i),\ P(i,j,k,l) \\
sup_{2,1}(i,j) & :- & m\_P(i),\ a_1(i,j) \\
sup_{2,2}(i,k) & :- & sup_{2,1}(i,j),\ a_2(j,k) \\
sup_{2,3}(i,k,l,m) & :- & sup_{2,2}(i,k)\ a_3(l,m) \\
sup_{3,1}(i,j) & :- & m\_P(i),\ a_1(i,j) \\
sup_{3,2}(i,k,n,o) & :- & sup_{3,1}(i,j),\ P(j,k,n,o) \\
sup_{3,3}(i,l,n,o) & :- & sup_{3,2}(i,k,n,o),\ a_2(k,l) \\
sup_{3,4}(i,l,m,o) & :- & sup_{3,3}(i,l,n,o),\ a_3(m,n) \\
sup_{4,1}(i,j,k,l) & :- & m\_Q(i,k,l),\ b_1(i,j) \\
sup_{4,2}(i,k,l) & :- & sup_{4,1}(i,j,k,l),\ b_2(j,k) \\
sup_{4,3}(i,k,l,m) & :- & sup_{4,2}(i,k,l),\ b_3(l,m) \\
sup_{5,1}(i,j,l,m) & :- & m\_Q(i,l,m),\ b_1(i,j) \\
sup_{5,2}(i,j,k,l,m) & :- & sup_{5,1}(i,j,l,m),\ b_2(k,l) \\
sup_{5,3}(i,j,k,l,m,n) & :- & sup_{5,2}(i,j,k,l,m),\ b_3(m,n) \\
sup_{5,4}(i,l,m,o) & :- & sup_{5,3}(i,j,k,l,m,n),\ Q(j,k,n,o) \\
\\
S(i,m) & :- & sup_{1,1}(i,j,k,l),\ Q(j,k,l,m) \\
P(i,k,l,n) & :- & sup_{2,3}(i,k,l,m),\ a_4(m,n) \\
P(i,l,m,p) & :- & sup_{3,4}(i,l,m,o),\ a_4(o,p) \\
Q(i,k,l,n) & :- & sup_{4.3}(i,k,l,m),\ b_4(m,n) \\
Q(i,l,m,p) & :- & sup_{5,4}(i,l,m,o),\ b_4(o,p)
\end{array}
$$

$a_1 a_2 a_3 a_4$

$$
\begin{array}{c}
a_1(0,1) \\
a_2(1,2) \\
a_3(2,3) \\
a_4(3,4) \\
m\_S(0)
\end{array}
$$

$$
\begin{array}{c}
sup_{1,1}(0,2,2,4)\ sup_{2,1}(0,1)\ sup_{2,2}(0,2) \\
sup_{2,3}(0,2,2,3)\ sup_{3,1}(0,1) \\
\\
m\_S(0) \quad m\_P(0) \quad m\_Q(2,2,4) \\
m\_P(1) \\
\\
P(0,2,2,4)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
m\_P(i) & :- & m\_S(i) \\
m\_Q(j,k,l) & :- & sup_{1,1}(i,j,k,l) \\
m\_P(j) & :- & sup_{3,1}(i,j) \\
m\_Q(j,k,n) & :- & sup_{5,3}(i,j,k,l,n) \\
\\
sup_{1,1}(i,j,k,l) & :- & m\_S(i),\ P(i,j,k,l) \\
sup_{2,1}(i,j) & :- & m\_P(i),\ a_1(i,j) \\
sup_{2,2}(i,k) & :- & sup_{2,1}(i,j),\ a_2(j,k) \\
sup_{2,3}(i,k,l,m) & :- & sup_{2,2}(i,k)\ a_3(l,m) \\
sup_{3,1}(i,j) & :- & m\_P(i),\ a_1(i,j) \\
sup_{3,2}(i,k,n,o) & :- & sup_{3,1}(i,j),\ P(j,k,n,o) \\
sup_{3,3}(i,l,n,o) & :- & sup_{3,2}(i,k,n,o),\ a_2(k,l) \\
sup_{3,4}(i,l,m,o) & :- & sup_{3,3}(i,l,n,o),\ a_3(m,n) \\
sup_{4,1}(i,j,k,l) & :- & m\_Q(i,k,l),\ b_1(i,j) \\
sup_{4,2}(i,k,l) & :- & sup_{4,1}(i,j,k,l),\ b_2(j,k) \\
sup_{4,3}(i,k,l,m) & :- & sup_{4,2}(i,k,l),\ b_3(l,m) \\
sup_{5,1}(i,j,l,m) & :- & m\_Q(i,l,m),\ b_1(i,j) \\
sup_{5,2}(i,j,k,l,m) & :- & sup_{5,1}(i,j,l,m),\ b_2(k,l) \\
sup_{5,3}(i,j,k,l,m,n) & :- & sup_{5,2}(i,j,k,l,m),\ b_3(m,n) \\
sup_{5,4}(i,l,m,o) & :- & sup_{5,3}(i,j,k,l,m,n),\ Q(j,k,n,o) \\
\\
S(i,m) & :- & sup_{1,1}(i,j,k,l),\ Q(j,k,l,m) \\
P(i,k,l,n) & :- & sup_{2,3}(i,k,l,m),\ a_4(m,n) \\
P(i,l,m,p) & :- & sup_{3,4}(i,l,m,o),\ a_4(o,p) \\
Q(i,k,l,n) & :- & sup_{4.3}(i,k,l,m),\ b_4(m,n) \\
Q(i,l,m,p) & :- & sup_{5,4}(i,l,m,o),\ b_4(o,p)
\end{array}
$$

$a_1 a_2 a_3 a_4$

$a_1(0,1)$
$a_2(1,2)$
$a_3(2,3)$
$a_4(3,4)$
$m\_S(0)$

$sup_{1,1}(0,2,2,4)$ $sup_{2,1}(0,1)$ $sup_{2,2}(0,2)$
$sup_{2,3}(0,2,2,3)$ $sup_{3,1}(0,1)$

$m\_S(0)$ $m\_P(0)$ $m\_Q(2,2,4)$
$m\_P(1)$

$P(0,2,2,4)$

The origin of the problem

$$
\begin{array}{rcl}
S^{bf}(i,m) & :- & P^{bfff}(i,j,k,l),\ Q^{bbbf}(j,k,l,m) \\
P^{bfff}(i,k,l,n) & :- & a_1^{bf}(i,j),\ a_2^{bf}(j,k),\ a_3^{ff}(l,m),\ a_4^{bf}(m,n) \\
P^{bfff}(i,l,m,p) & :- & a_1^{bf}(i,j),\ P^{bfff}(j,k,n,o),\ a_2^{bf}(k,l),\ a_3^{fb}(m,n),\ a_4^{bf}(o,p) \\
Q^{bbbf}(i,k,l,n) & :- & a_1^{bf}(i,j),\ b_2^{bb}(j,k),\ b_3^{bf}(l,m),\ b_4^{bb}(m,n) \\
Q^{bbbf}(i,l,m,p) & :- & b_1^{bf}(i,j),\ b_2^{fb}(k,l),\ b_3^{fb}(m,n),\ Q^{bbbf}(j,k,n,o),\ b_4^{bf}(o,p)
\end{array}
$$

- we need that variables do not shuffle too much
- we need to use new predicates to ensure the prefix correctness

The origin of the problem

$$
\begin{array}{rcl}
S^{bf}(i,m) & :- & P^{bfff}(i,j,k,l),\ Q^{bbbf}(j,k,l,m) \\
P^{bfff}(i,k,l,n) & :- & a_1^{bf}(i,j),\ a_2^{bf}(j,k),\ a_3^{ff}(l,m),\ a_4^{bf}(m,n) \\
P^{bfff}(i,l,m,p) & :- & a_1^{bf}(i,j),\ P^{bfff}(j,k,n,o),\ a_2^{bf}(k,l),\ a_3^{fb}(m,n),\ a_4^{bf}(o,p) \\
Q^{bbbf}(i,k,l,n) & :- & a_1^{bf}(i,j),\ b_2^{bb}(j,k),\ b_3^{bf}(l,m),\ b_4^{bb}(m,n) \\
Q^{bbbf}(i,l,m,p) & :- & b_1^{bf}(i,j),\ b_2^{fb}(k,l),\ b_3^{fb}(m,n),\ Q^{bbbf}(j,k,n,o),\ b_4^{bf}(o,p)
\end{array}
$$

- we need that variables do not shuffle too much
- we need to use new predicates to ensure the prefix correctness

The origin of the problem

$$
\begin{array}{rcl}
S^{bf}(i,m) & :- & P^{bfff}(i,j,k,l),\ Q^{bbbf}(j,k,l,m) \\
P^{bfff}(i,k,l,n) & :- & a_1^{bf}(i,j),\ a_2^{bf}(j,k),\ a_3^{ff}(l,m),\ a_4^{bf}(m,n) \\
P^{bfff}(i,l,m,p) & :- & a_1^{bf}(i,j),\ P^{bfff}(j,k,n,o),\ a_2^{bf}(k,l),\ a_3^{fb}(m,n),\ a_4^{bf}(o,p) \\
Q^{bbbf}(i,k,l,n) & :- & a_1^{bf}(i,j),\ b_2^{bb}(j,k),\ b_3^{bf}(l,m),\ b_4^{bb}(m,n) \\
Q^{bbbf}(i,l,m,p) & :- & b_1^{bf}(i,j),\ b_2^{fb}(k,l),\ b_3^{fb}(m,n),\ Q^{bbbf}(j,k,n,o),\ b_4^{bf}(o,p)
\end{array}
$$

- we need that variables do not shuffle too much
- we need to use new predicates to ensure the prefix correctness

Ordered grammars

Given an MCFG $G = (N, T, P, S)$, we build an ordered MCFG $G' = (N', T, P', S)$ such that:

- if A is in N and has rank n, for all permutation σ of $[1; n]$, A_σ is in R',
- if
$$A(s_1, \ldots, s_n) \leftarrow B_1(x_1^1, \ldots, x_{k_1}^1), \ldots, B_m(x_1^m, \ldots, x_{k_m}^m)$$
is in P then
$$A_\sigma(s_{\sigma(1)}, \ldots, s_{\sigma(n)}) \leftarrow B_{1,\tau_1}(x^1_{\tau_1(1)}, \ldots, x^1_{\tau_1(k_1)}), \ldots, B_{m,\tau_m}(x^m_{\tau_m(1)}, \ldots, x^m_{\tau_m(k_m)})$$
so that for every i, $x^i_{\tau_i(1)}, \ldots, x^i_{\tau_i(k_i)}$ appears in that order in $s_{\sigma(1)} \ldots s_{\sigma(n)}$.

NB: this can make the size of the grammar explode:

$$\begin{array}{rcl} S(x_1 \ldots x_n) & \leftarrow & A(x_1, \ldots, x_n) \\ A(a_1, \ldots, a_n) & \leftarrow & \end{array} \qquad \begin{array}{rcl} A(x_n, x_1, \ldots, x_{n-1}) & \leftarrow & A(x_1, \ldots, x_n) \\ A(x_2, x_1, x_3 \ldots, x_n) & \leftarrow & A(x_1, \ldots, x_n) \end{array}$$

Ordered grammars

Given an MCFG $G = (N, T, P, S)$, we build an ordered MCFG $G' = (N', T, P', S)$ such that:

- if A is in N and has rank n, for all permutation σ of $[1; n]$, A_σ is in R',
- if

$$A(s_1, \ldots, s_n) \leftarrow B_1(x_1^1, \ldots, x_{k_1}^1), \ \ldots, \ B_m(x_1^m, \ldots, x_{k_m}^m)$$

  is in P then

$$A_\sigma(s_{\sigma(1)}, \ldots, s_{\sigma(n)}) \leftarrow B_{1,\tau_1}(x^1_{\tau_1(1)}, \ldots, x^1_{\tau_1(k_1)}), \ldots, B_{m,\tau_m}(x^m_{\tau_m(1)}, \ldots, x^m_{\tau_m(k_m)})$$

  so that for every i, $x^i_{\tau_i(1)}, \ldots, x^i_{\tau_i(k_i)}$ appears in that order in $s_{\sigma(1)} \ldots s_{\sigma(n)}$.

NB: this can make the size of the grammar explode:

$$\begin{array}{rcl} S(x_1 \ldots x_n) & \leftarrow & A(x_1, \ldots, x_n) \\ A(a_1, \ldots, a_n) & \leftarrow & \end{array} \qquad \begin{array}{rcl} A(x_n, x_1, \ldots, x_{n-1}) & \leftarrow & A(x_1, \ldots, x_n) \\ A(x_2, x_1, x_3 \ldots, x_n) & \leftarrow & A(x_1, \ldots, x_n) \end{array}$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P_1(i, j),\ P(i, j, k, l),\ Q_1(j, k),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P_1(j, k),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q_1(j, k),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P_1(i, j),\ P(i, j, k, l),\ Q_1(j, k),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P_1(j, k),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q_1(j, k),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P_1(i, j),\ P(i, j, k, l),\ Q_1(j, k),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P_1(j, k),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q_1(j, k),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p) \\
P_1(i, k) & :- & a_1(i, j),\ a_2(j, k)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P_1(i, j),\ P(i, j, k, l),\ Q_1(j, k),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P_1(j, k),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q_1(j, k),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p) \\
P_1(i, k) & :- & a_1(i, j),\ a_2(j, k)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{aligned}
S(x_1 y_1 x_2 y_2) &\leftarrow P(x_1, x_2),\ Q(y_1, y_2)\\
P(a_1 x_1 a_2, a_3 x_2 a_4) &\leftarrow P(x_1, x_2)\\
P(a_1 a_2, a_3 a_4) &\leftarrow\\
Q(b_1 y_1 b_2, b_3 y_2 b_4) &\leftarrow Q(y_1, y_2)\\
Q(b_1 b_2, b_3 b_4) &\leftarrow
\end{aligned}
$$

$$
\begin{aligned}
S(i, m) &:- P(i, j, k, l),\ Q(j, k, l, m)\\
P(i, k, l, n) &:- a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n)\\
P(i, l, m, p) &:- a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p)\\
Q(i, k, l, n) &:- a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n)\\
Q(i, l, m, p) &:- b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{aligned}
$$

$$
\begin{aligned}
S(i, m) &:- P_1(i, j),\ P(i, j, k, l),\ Q_1(j, k),\ Q(j, k, l, m)\\
P(i, k, l, n) &:- aux_1(i, k),\ a_3(l, m),\ a_4(m, n)\\
P(i, l, m, p) &:- a_1(i, j),\ P_1(j, k),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p)\\
Q(i, k, l, n) &:- a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n)\\
Q(i, l, m, p) &:- b_1(i, j),\ Q_1(j, k),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)\\
P_1(i, k) &:- aux_1(i, k)\\
aux_1(i, k) &:- a_1(i, j),\ a_2(j, k)
\end{aligned}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P_1(i, j),\ P(i, j, k, l),\ Q_1(j, k),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & aux_1(i, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P_1(j, k),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q_1(j, k),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p) \\
P_1(i, k) & :- & aux_1(i, k) \\
aux_1(i, k) & :- & a_1(i, j),\ a_2(j, k)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1y_1x_2y_2) & \leftarrow & P(x_1,x_2),\ Q(y_1,y_2) \\
P(a_1x_1a_2,\ a_3x_2a_4) & \leftarrow & P(x_1,x_2) \\
P(a_1a_2,\ a_3a_4) & \leftarrow & \\
Q(b_1y_1b_2,\ b_3y_2b_4) & \leftarrow & Q(y_1,y_2) \\
Q(b_1b_2,\ b_3b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i,m) & :- & P(i,j,k,l),\ Q(j,k,l,m) \\
P(i,k,l,n) & :- & a_1(i,j),\ a_2(j,k),\ a_3(l,m),\ a_4(m,n) \\
P(i,l,m,p) & :- & a_1(i,j),\ P(j,k,n,o),\ a_2(k,l),\ a_3(m,n),\ a_4(o,p) \\
Q(i,k,l,n) & :- & a_1(i,j),\ b_2(j,k),\ b_3(l,m),\ b_4(m,n) \\
Q(i,l,m,p) & :- & b_1(i,j),\ Q(j,k,n,o),\ b_2(k,l),\ b_3(m,n),\ b_4(o,p)
\end{array}
$$

$$
\begin{array}{rcl}
S(i,m) & :- & P_1(i,j),\ P(i,j,k,l),\ Q_1(j,k),\ Q(j,k,l,m) \\
P(i,k,l,n) & :- & aux_1(i,k),\ a_3(l,m),\ a_4(m,n) \\
P(i,l,m,p) & :- & a_1(i,j),\ P_1(j,k),\ P(j,k,n,o),\ a_2(k,l),\ a_3(m,n),\ a_4(o,p) \\
Q(i,k,l,n) & :- & a_1(i,j),\ b_2(j,k),\ b_3(l,m),\ b_4(m,n) \\
Q(i,l,m,p) & :- & b_1(i,j),\ Q_1(j,k),\ Q(j,k,n,o),\ b_2(k,l),\ b_3(m,n),\ b_4(o,p) \\
P_1(i,k) & :- & aux_1(i,k) \\
aux_1(i,k) & :- & a_1(i,j),\ a_2(j,k) \\
P_1(i,l) & :- & a_1(i,j),\ P_1(j,k),\ a_2(k,l)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}$$

$$\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}$$

$$\begin{array}{rcl}
S(i, m) & :- & P_1(i, j),\ P(i, j, k, l),\ Q_1(j, k),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & aux_1(i, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P_1(j, k),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q_1(j, k),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p) \\
P_1(i, k) & :- & aux_1(i, k) \\
aux_1(i, k) & :- & a_1(i, j),\ a_2(j, k) \\
P_1(i, l) & :- & a_1(i, j),\ P_1(j, k),\ a_2(k, l)
\end{array}$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P_1(i, j),\ P(i, j, k, l),\ Q_1(j, k),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & aux_1(i, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & aux_2(i, j, k, l),\ P(j, k, n, o),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q_1(j, k),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p) \\
P_1(i, k) & :- & aux_1(i, k) \\
aux_1(i, k) & :- & a_1(i, j),\ a_2(j, k) \\
P_1(i, l) & :- & aux_2(i, j, k, l) \\
aux_2(i, j, k, l) & :- & a_1(i, j),\ P_1(j, k),\ a_2(k, l)
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P(i, j, k, l),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & a_1(i, j),\ a_2(j, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & a_1(i, j),\ P(j, k, n, o),\ a_2(k, l),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & a_1(i, j),\ b_2(j, k),\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & b_1(i, j),\ Q(j, k, n, o),\ b_2(k, l),\ b_3(m, n),\ b_4(o, p)
\end{array}
$$

$$
\begin{array}{rcl}
S(i, m) & :- & P_1(i, j),\ P(i, j, k, l),\ Q_1(j, k),\ Q(j, k, l, m) \\
P(i, k, l, n) & :- & aux_1(i, k),\ a_3(l, m),\ a_4(m, n) \\
P(i, l, m, p) & :- & aux_2(i, j, k, l),\ P(j, k, n, o),\ a_3(m, n),\ a_4(o, p) \\
Q(i, k, l, n) & :- & \color{red}{aux_3(i, k)},\ b_3(l, m),\ b_4(m, n) \\
Q(i, l, m, p) & :- & \color{red}{aux_4(i, j, k, l)},\ Q(j, k, n, o),\ b_3(m, n),\ b_4(o, p) \\
P_1(i, k) & :- & aux_1(i, k) \\
aux_1(i, k) & :- & a_1(i, j),\ a_2(j, k) \\
P_1(i, l) & :- & aux_2(i, j, k, l) \\
aux_2(i, j, k, l) & :- & a_1(i, j),\ P_1(j, k),\ a_2(k, l) \\
\color{red}{Q_1(i, k)} & \color{red}{:-} & \color{red}{aux_3(i, k)} \\
\color{red}{aux_3(i, k)} & \color{red}{:-} & \color{red}{a_1(i, j),\ a_2(j, k)} \\
\color{red}{Q_1(i, l)} & \color{red}{:-} & \color{red}{aux_4(i, j, k, l)} \\
\color{red}{aux_4(i, j, k, l)} & \color{red}{:-} & \color{red}{b_1(i, j),\ Q_1(j, k),\ b_2(k, l)}
\end{array}
$$

MCFG parsing and Datalog: an example by Kanazawa

$$
\begin{array}{rcl}
S(x_1 y_1 x_2 y_2) & \leftarrow & P(x_1, x_2),\ Q(y_1, y_2) \\
P(a_1 x_1 a_2, a_3 x_2 a_4) & \leftarrow & P(x_1, x_2) \\
P(a_1 a_2, a_3 a_4) & \leftarrow & \\
Q(b_1 y_1 b_2, b_3 y_2 b_4) & \leftarrow & Q(y_1, y_2) \\
Q(b_1 b_2, b_3 b_4) & \leftarrow &
\end{array}
$$

$$
\begin{array}{rcl}
S(i,m) & :- & P(i,j,k,l),\ Q(j,k,l,m) \\
P(i,k,l,n) & :- & a_1(i,j),\ a_2(j,k),\ a_3(l,m),\ a_4(m,n) \\
P(i,l,m,p) & :- & a_1(i,j),\ P(j,k,n,o),\ a_2(k,l),\ a_3(m,n),\ a_4(o,p) \\
Q(i,k,l,n) & :- & a_1(i,j),\ b_2(j,k),\ b_3(l,m),\ b_4(m,n) \\
Q(i,l,m,p) & :- & b_1(i,j),\ Q(j,k,n,o),\ b_2(k,l),\ b_3(m,n),\ b_4(o,p)
\end{array}
$$

$$
\begin{array}{rcl}
S(i,m) & :- & P_1(i,j),\ P(i,j,k,l),\ Q_1(j,k),\ Q(j,k,l,m) \\
P(i,k,l,n) & :- & aux_1(i,k),\ a_3(l,m),\ a_4(m,n) \\
P(i,l,m,p) & :- & aux_2(i,j,k,l),\ P(j,k,n,o),\ a_3(m,n),\ a_4(o,p) \\
Q(i,k,l,n) & :- & aux_3(i,k),\ b_3(l,m),\ b_4(m,n) \\
Q(i,l,m,p) & :- & aux_4(i,j,k,l),\ Q(j,k,n,o),\ b_3(m,n),\ b_4(o,p) \\
P_1(i,k) & :- & aux_1(i,k) \\
aux_1(i,k) & :- & a_1(i,j),\ a_2(j,k) \\
P_1(i,l) & :- & aux_2(i,j,k,l) \\
aux_2(i,j,k,l) & :- & a_1(i,j),\ P_1(j,k),\ a_2(k,l) \\
Q_1(i,k) & :- & aux_3(i,k) \\
aux_3(i,k) & :- & a_1(i,j),\ a_2(j,k) \\
Q_1(i,l) & :- & aux_4(i,j,k,l) \\
aux_4(i,j,k,l) & :- & b_1(i,j),\ Q_1(j,k),\ b_2(k,l)
\end{array}
$$

$$
\begin{array}{rcl}
S^{bf}(i,m) & :- & P_1^{bf}(i,j),\ Q_1^{bf}(j,k),\ P^{bbbf}(i,j,k,l),\ Q^{bbbf}(j,k,l,m) \\
P^{bbbf}(i,k,l,n) & :- & aux_1^{bb}(i,k),\ a_3^{bf}(l,m),\ a_4^{bf}(m,n) \\
P^{bbbf}(i,l,m,p) & :- & aux_2^{bffb}(i,j,k,l),\ a_3^{bf}(m,n),\ P^{bbbf}(j,k,n,o),\ a_4^{bf}(o,p) \\
Q^{bbbf}(i,k,l,n) & :- & aux_3^{bf}(i,k),\ b_3^{bf}(l,m),\ b_4^{bf}(m,n) \\
Q^{bbbf}(i,l,m,p) & :- & aux_4^{bffb}(i,j,k,l),\ b_3^{bf}(m,n),\ Q^{bbbf}(j,k,n,o),\ b_4^{bf}(o,p) \\
P_1^{bf}(i,k) & :- & aux_1^{bf}(i,k) \\
aux_1^{bf}(i,k) & :- & a_1^{bf}(i,j),\ a_2^{bf}(j,k) \\
P_1^{bf}(i,l) & :- & aux_2^{bffb}(i,j,k,l) \\
aux_2^{bffb}(i,j,k,l) & :- & a_1^{bf}(i,j),\ P_1^{bf}(j,k),\ a_2^{bb}(k,l) \\
Q_1^{bf}(i,k) & :- & aux_3^{bf}(i,k) \\
aux_3^{bf}(i,k) & :- & a_1^{bf}(i,j),\ a_2^{bf}(j,k) \\
Q_1^{bf}(i,l) & :- & aux_4^{bffb}(i,j,k,l) \\
aux_4^{bffb}(i,j,k,l) & :- & b_1^{bf}(i,j),\ Q_1^{bf}(j,k),\ b_2^{bb}(k,l)
\end{array}
$$

Extending the technique

- This technique can easily be extended to PMCFG (Ball master thesis 2009).
- It can be used to analyze regular languages.
- Hopefully other kinds of program transformations can yield other parsing strategies (left corner, generalized LR...)

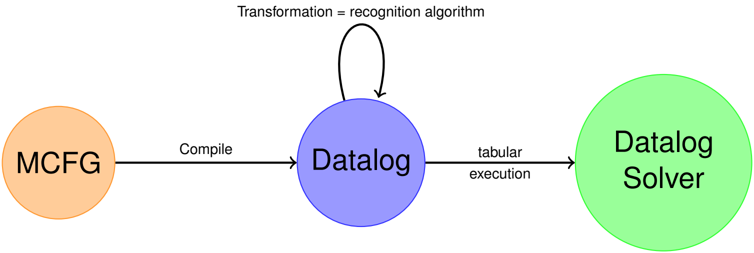
Extending the technique

- This technique can easily be extended to PMCFG (Ball master thesis 2009).
- It can be used to analyze regular languages.
- Hopefully other kinds of program transformations can yield other parsing strategies (left corner, generalized LR. . .)

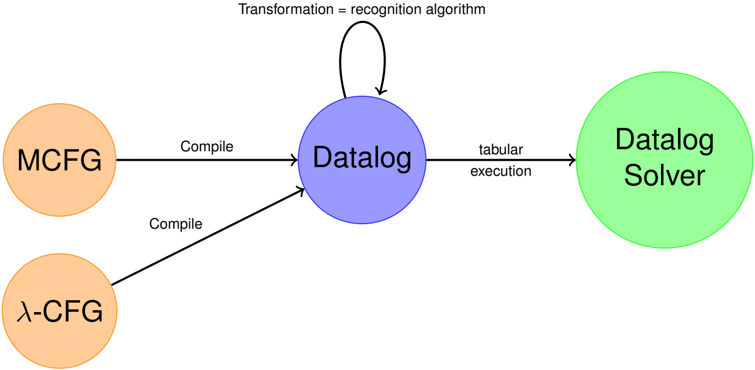
Extending the technique

- This technique can easily be extended to PMCFG (Ball master thesis 2009).
- It can be used to analyze regular languages.
- Hopefully other kinds of program transformations can yield other parsing strategies (left corner, generalized LR. . .)

Interest of the technique



Interest of the technique



Interest of the technique